Heriot-Watt University

Developing Semantic Pathway Comparison Methods for Systems Biology

Jonas Gamalielsson

September 5, 2009

Submitted for the degree of
Doctor of Philosophy in Computer Science
on completion of research in the
Department of Computer Science,
School of Mathematical and Computer Sciences.

Abstract

Systems biology is an emerging multi-disciplinary field in which the behaviour of complex biological systems is studied by considering the interaction of many cellular and molecular constituents rather than using a "traditional" reductionist approach where constituents are studied individually. Systems are often studied over time with the ultimate goal of developing models which can be used to understand and predict complex biological processes, such as human diseases. To support systems biology, a large number of biological pathways are being derived for many different organisms, and these are stored in various databases. This pathway collection presents an opportunity to compare and contrast pathways, and to utilise the knowledge they represent. This thesis presents some of the first algorithms that are designed to explore this opportunity. It is argued that the methods will be useful to biologists in order to assess the biological plausibility of derived pathways, compare different biological pathways for semantic similarities, and to derive putative pathways that are semantically similar to documented biological pathways. The methods will therefore extend the systems biology toolbox that biologists can use to make new biological discoveries.

Acknowledgements

First I would like to thank my supervisors David W. Corne and Björn Olsson. David for providing useful advice and feedback on my work and for arranging the viva. Björn for being a valuable source of ideas and feedback, and for being my fellow co-writer in many research papers. Thanks also to my former supervisor at Exeter, Ajit Narayanan, for always challenging me during our research chats. Additional thanks to my thesis examiners (Dr. David Robertson and Dr. Albert Burger) for taking their time to assess and give feedback on my work. Furthermore, I would like to thank my dear colleagues Angelica, Jane, Kim, Simon and Zelmina for always being very good friends and for supporting me. A special thanks goes to my beloved wife Rattana and my son Johan. Thanks for being there for me and providing a normal life style behind the scenes of postgraduate studies. I would also like to thank the University of Skövde for supporting me financially during my studies. This research was also supported by project grant number 2003/0215 from the Knowledge Foundation, and by the Information Fusion Research Program (University of Skövde, Sweden) under grant 2003/0104 in partnership with the Knowledge Foundation (URL: http://www.infofusion.se)

ACADEMIC REGISTRY
Research Thesis Submission

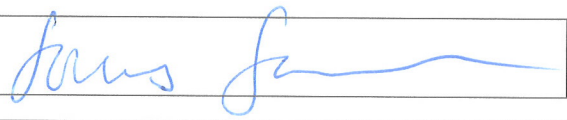
Name:	Jonas Gamalielsson		
School/PGI:	School of Mathematical and Computer Sciences		
Version: *(i.e. First, Resubmission, Final)*	Final	Degree Sought (Award **and** Subject area)	Doctor of Philosophy in Computer Science

Declaration

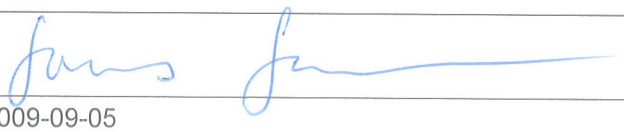
In accordance with the appropriate regulations I hereby submit my thesis and I declare that:

1) the thesis embodies the results of my own work and has been composed by myself
2) where appropriate, I have made acknowledgement of the work of others and have made reference to work carried out in collaboration with other persons
3) the thesis is the correct version of the thesis for submission and is the same version as any electronic versions submitted*.
4) my thesis for the award referred to, deposited in the Heriot-Watt University Library, should be made available for loan or photocopying and be available via the Institutional Repository, subject to such conditions as the Librarian may require
5) I understand that as a student of the University I am required to abide by the Regulations of the University and to conform to its discipline.

* *Please note that it is the responsibility of the candidate to ensure that the correct version of the thesis is submitted.*

Signature of Candidate:		Date:	2009-09-05

Submission

Submitted By *(name in capitals)*:	JONAS GAMALIELSSON
Signature of Individual Submitting:	
Date Submitted:	2009-09-05

For Completion in Academic Registry

Received in the Academic Registry by *(name in capitals)*:			
Method of Submission *(Handed in to Academic Registry; posted through internal/external mail)*:			
*E-thesis Submitted (**mandatory from January 2009**)*			
Signature:		Date:	

Please note this form should bound into the submitted thesis.

Updated February 2008, November 2008, February 2009

Contents

List of Tables

List of Figures

Dissemination

Gamalielsson, J. and Olsson, B. (2008). Gene Ontology-Based Semantic Alignment of Biological Pathways by Evolutionary Search, *Journal of Bioinformatics and Computational Biology*, **6**(4):825-842.

Gamalielsson, J. and Olsson, B. (2008). GOSAP: Gene Ontology Based Semantic Alignment of Biological Pathways, *International Journal of Bioinformatics Research and Applications*, **4**(3):274-294.

Gamalielsson, J. and Olsson, B. (2007). EGOSAP: Evolutionary Gene Ontology Based Semantic Alignment of Biological Pathways, *Proceedings of 3rd Moscow Conference on Computational Molecular Biology (MCCMB07)*, Moscow, Russia, July 27-30, 2007.

Gamalielsson, J. , Nilsson, P and Olsson, B. (2006). A GO-based Method for Assessing the Biological Plausibility of Regulatory Hypotheses. In Alexandrov, V. N., van Albada, G. D., Sloot, M. A., and Dongarra, J. (eds.), *Proceedings of ICCS 2006: 6th International Conference on Computational Science*, LNCS 3992: 879-886. Springer-Verlag.

In the papers above, Jonas Gamalielsson has contributed with the text in itself, background research, method development, experimental design, results, analysis and discussion. Björn Olsson has contributed with feedback on the work by Jonas Gamalielsson, proofreading and suggestions for stilistic changes to the text. Patric Nilsson has contributed with proofreading.

In addition to the papers, a provisional patent application titled *Biological plausibility determination utilizing the matching of regulatory hypotheses to templates* (US60/594,234) was filed, where the author is first inventor.

Research experience

Research experience of the author, which is not directly related to this doctoral thesis, is discussed here. During an earlier employment at Allgon AB in Åkersberga (1995-2001) he worked as an RF engineer in a research group. Tasks comprised electromagnetic testing and development of new cellular telephone antenna concepts, but also development of software for instrument control, data acquisition, data analysis and electromagnetic simulations. The author is first inventor of two patented antenna devices; *Dual band antenna*(PCT WO 01/11899) and *Antenna device for a hand-portable radio communication unit* (PCT WO 99/03166, US pat 6388626). A paper was written about the software *CGAMAS* that was developed for analysis of microwave data files (Rowell and Gamalielsson 1997). In collaboration with Allgon AB, a BSc thesis was written on the applicability of a genetic algorithm when optimising the performance of an integrated antenna for cellular telephones (Gamalielsson 2000). At the university of Skövde, the author has performed research on development and evaluation of a method for *ab initio* prediction of protein structures using simplified off-lattice models and evolutionary algorithms. The work resulted in a MSc thesis (Gamalielsson 2001), a conference paper (Gamalielsson and Olsson 2002) and also a book chapter (Gamalielsson and Olsson 2005a). Additionally, research on robustness of clustering algorithms applied to gene expression data has been performed by the author, where a robustness index was proposed and gene expression datasets were evaluated (Gamalielsson and Olsson 2004). The author has also contributed in a collaborative research effort on the application of artificial neural networks and genetic algorithms to gene expression data (Narayanan et al. 2004, Narayanan et al. 2005). The author was significantly involved in research where methods were developed which combine semantic and topological analysis of protein interaction networks with the aim to identify functional modules (Lubovac et al. 2005a, Lubovac et al. 2005b, Lubovac et al. 2006a, Lubovac et al. 2006b, Lubovac et al. 2007). Furthermore, he has contributed in research aiming to relate methods from bioinformatics and systems biology to the field of information fusion (Synnergren et al. 2007, Synnergren et al. 2008). The author has also completed referee assignments for the journals *IEEE Transactions on Computational Biology and Bioinformatics*, and *Neurocomputing*.

Chapter 1

Introduction

Systems biology is a research field where an understanding of the interaction of all cellular and molecular constituents, such as genes and proteins, is sought (Kirschner 2005, Liu 2005). In many cases, systems are studied over time (Aderem 2005), with the goal to develop models which can be used to understand and predict complex biological processes like human diseases (Butcher et al. 2004). In order to obtain such an understanding, biological systems can be studied at different levels of detail and composition. Such an abstraction framework is described in Oltvai and Barabasi (2002), where the aforementioned levels are organised as a pyramid. Individual constituents, such as genes and proteins, are studied at the bottom level. This is often referred to as the reductionistic approach (Ahn et al. 2006) where much effort is spent on seeking detailed knowledge of a limited number of constituents. So far, the interplay between constituents is more or less ignored. At higher levels in the pyramid, knowledge is seeked about the interaction between constituents. Such knowledge is typically represented by different kinds of biological graphs, which can be derived by combining the results from a set of reductionistic experiments. It is also common that large scale experimental platforms such as DNA microarrays (Schena et al. 1995) are used to derive graphs describing the interplay between a large number of genes or proteins (Werhli et al. 2006, Bansal et al. 2007). There is a need for powerful algorithms for analysis of biological graphs, because the amount and size of graphs is increasing rapidly due to the explosion in number of large scale experiments performed (Koyutürk et al. 2004). One class of such algorithms can be defined as graph comparison algorithms, where

graphs are compared in order to find similar subgraphs. In turn, we can define different sub-classes of graph comparison algorithms for the analysis of biological graphs, which have different purposes. In the following section and its sub-sections, we present problem scenarios for which we have developed methods for graph comparison.

1.1 Problem scenarios

We have identified three different problem scenarios that are all part of systems biology, and for which there is a need for powerful methods for graph comparison. For each problem scenario, we have developed a specific kind of graph comparison method. All three methods are similar in that they use Gene Ontology (GO)-based (Ashburner et al. 2000) semantic similarity as means to compare gene products with each other. Since it is more common in the literature to refer to biological graphs as biological networks or pathways, we will prefer to use that terminology from now on.

1.1.1 Assessing reverse engineered regulatory networks

Forward engineering is the "classic" way of modelling biological networks, where putative networks are wired by hand and coupled differential equations are used to simulate the behaviour of different biological processes (Werhli et al. 2006). This top-down approach of modelling relies heavily on previously known molecular mechanisms, and also on subjective decisions made during modelling.

The idea of automatically deriving biological networks from data is appealing. The potential of using microarray gene expression data as input to so called reverse engineering algorithms has been explored by numerous researchers during the last decade. Reverse engineering is a bottom-up approach, which is about deriving the structure of some system by backwards reasoning using observations of the behaviour of the system (Hartemink 2005). This behaviour can be studied using recent large scale measurement technologies such as DNA microarrays (Werhli et al. 2006, Bansal et al. 2007). By letting the network emerge from data rather than data emerge from the network, the chance of finding novel interactions increases. Figure 1.1 illustrates the reverse engineering process. Sometimes a synthetic network is used to generate ex-

pression data, because real biological networks found in databases can be incomplete and microarray gene expression datasets can be imperfect in terms of experimental design (Bansal et al. 2007). The expression data is used as input to the reverse engineering algorithm. After the genetic network has been derived, it can be compared to the "gold standard" (either synthetic or real) network using e.g. a sensitivity and specificity analysis (Werhli et al. 2006, Bansal et al. 2007). Reverse engineering algorithms may also derive multiple models, each fitting the same data. The question then arises; which model is the biologically most relevant for the given data. Algorithms based only on gene expression data can not answer this question. However, additional data sources may help in assessing the biological plausibility of gene regulatory interactions. Examples of such data sources are binding site and transcription factor databases, databases of known regulatory pathways and databases describing the biological roles of genes and their products.

In this thesis we propose a GO-based method for assessing the biological plausibility of derived gene regulatory networks. Our approach derives semantic templates describing the molecular function of the gene products, as observed in binary interactions found in a set of known regulatory networks. Templates are assigned a score, where higher scores indicate more specific molecular functions. The molecular function of the gene products in interactions found in derived regulatory networks are subsequently matched to the templates. The score of template matches reflects the level of biological plausibility. A sorted list of the highest-scoring matches is presented for each derived regulatory interaction.

1.1.2 Aligning biological pathways

The era of large-scale biological experiments has resulted in a rapidly growing number of biological pathways (Yang and Sze 2007). Some have been assembled manually using results from large-scale or reductionistic experiments, often found in research papers or books. Other pathways have been computationally derived using different kinds of large-scale data. By performing an alignment between pathways A and B, as illustrated in figure 1.2, a pathway alignment is derived which may exhibit significant similarity between the two pathways. Additional data sources may also

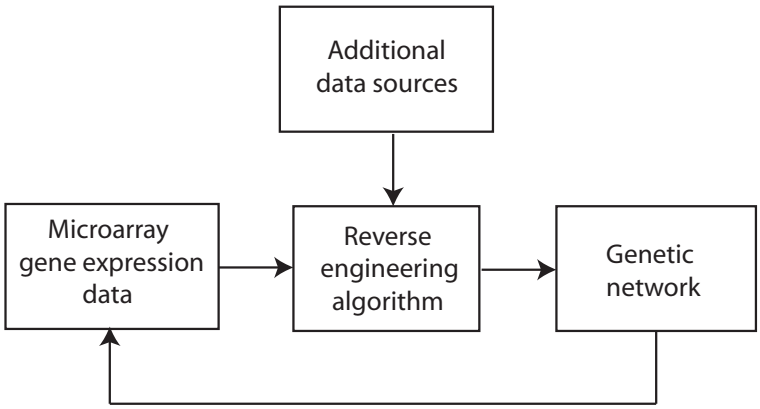


Figure 1.1: Reverse engineering gene regulatory networks.

serve as input to the pathway alignment algorithm, if the alignment shall be based on other properties than pathway topology and node labels. Examples of such data sources are DNA/protein sequence databases and data sources describing the semantic properties of the nodes. One semantic data source is the enzyme nomenclature of chemical reactions (Nomenclature Committee of the International Union of Biochemistry and Molecular Biology 1992), which is hierarchically organised. Another is the Gene Ontology, an abstraction hierarchy organised as a directed acyclic graph, which describes the molecular functions, biological processes and cellular components of gene products. By using such additional data sources it is possible to perform similarity-based approximate matching using the properties of the nodes, rather than performing exact matching using the node labels alone. Just as sequence alignments may help in the identification of evolutionary changes like insertions, deletions and substitutions (Durbin et al. 1998), pathway alignments may help in the identification of evolutionary events at the pathway level, such as gene duplication and divergence of function (Pinter et al. 2005). Pathway alignment algorithms can be used both to find intra-pathway, intra-species and inter-species similarities. Intra-pathway comparisons may be useful in order to find similar biological mechanisms within the same pathway. Intra-species comparison refers to the case when different pathways for the same organism are compared for similarities. This can be useful in order to understand how metabolism has evolved within a species (Pinter et al. 2005). The inter-species case is useful for studying the degree of evolutionary conservation between different species.

We propose a GO-based method for performing local alignment on paths extracted from two biological pathways where the nodes are gene products. A statistical test is used in order to detect significant alignments. With the proposed method, any kind of biological pathway can be used as long as the nodes represent gene products. The alignment output contains two parts; one "traditional" path alignment containing two paths of gene products, possibly including gaps, and one meta-alignment which contains the most specific GO-terms that the corresponding gene products have in common.

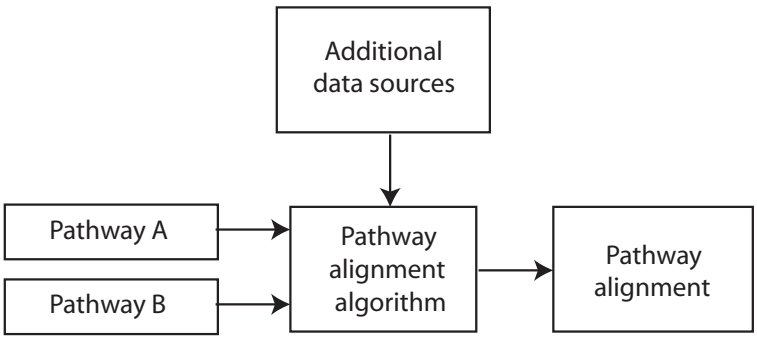


Figure 1.2: Aligning biological pathways.

1.1.3 Mapping derived genes onto pathways

A problem scenario related to biological pathway alignment is when a set of unordered genes or gene products of interest have been derived using e.g. microarray gene expression analysis, examples being differential analysis and clustering. Differential analysis yields a set of genes whose mRNA expression is significantly changed between two experimental conditions (Knudsen 2002). Clustering refers to the process of grouping genes into clusters, which are often derived using the similarity in shape of the genes' expression profiles over a set of experimental conditions. In this scenario it is of interest to investigate how genes from such gene sets can be mapped onto known biological pathways (Dahlquist et al. 2002). It might e.g. be the case that genes in the set are previously known to be participating in cancer-related pathways, and that the effects of the particular experiment might have something to do with the development of a particular kind of cancer. It should be pointed out that, in

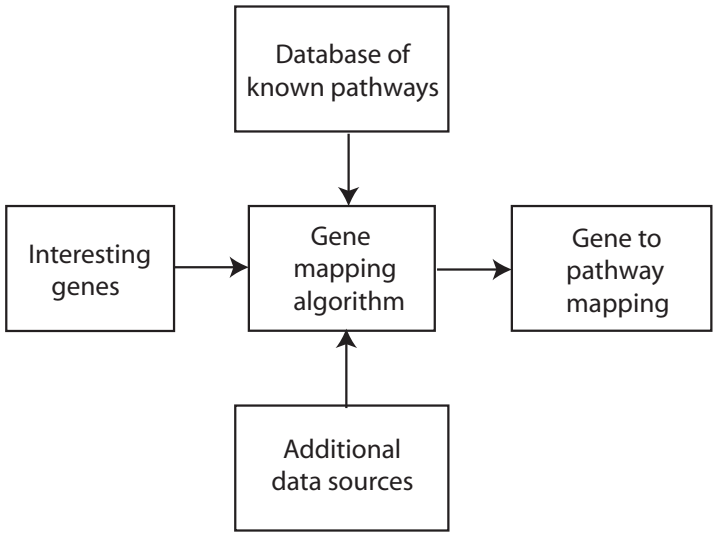


Figure 1.3: Mapping derived genes onto pathways.

general, not all genes in a set of differentially expressed genes or in a cluster, can be mapped to a specific pathway. Furthermore, other genes of interest, not being part of the particular experiment, could be added to the set of interesting genes. Figure 1.3 illustrates the process of mapping genes, or their products, onto pathways. The input is the derived set of interesting genes and a database of known pathways. By including additional data sources, e.g. for the characterisation of genes and their products, it will be possible to enrich the results or to perform approximate matching between genes in the "interesting set" and in the known pathways. Approximate matching is of interest since genes with different labels may still be very similar in terms of their biological roles (Koyutürk et al. 2004, Pinter et al. 2005). The output illustrates how the interesting genes are related to known pathways and genes in the known pathways.

Here, we propose a method for assembling paths of gene products from the set of interesting genes. The assembled paths are similar to paths present in the known pathways in terms of GO semantic similarity. In fact, the mapping is an alignment containing one path from the database of known pathways and one path assembled from the set of genes. The latter path may include gaps. A statistical test is used to detect significant alignments.

1.2 Contributions

In this thesis, three related methods are proposed for semantic comparison of biological pathways. The contributions of the methods are described in the following.

1. I contribute a novel method for distinguishing between plausible and implausible hypotheses in regulatory network construction, thereby improving the specificity of the regulatory network reconstruction process. The method generalises knowledge about gene products and their interactions using the molecular function terms of Gene Ontology (GO), rather than using the identity of gene products.

2. I contribute a novel approach to semantic local alignment of gene product paths extracted from biological pathways. The Gene Ontology, annotation databases and pathway databases serve as input data to the method. The method generalises about any kind of gene product, i.e. any kind of biological pathway can be aligned where the nodes are gene products, including gene/protein regulatory pathways, signalling pathways, and metabolic enzyme-to-enzyme pathways.

3. I find empirical evidence to suggest that the richer semantic description of gene products obtained by combining the function-, process- and component ontologies of GO in similarity calculations, increases the sensitivity and specificity of the path alignment process, compared to the case where any of the ontologies is used individually.

4. I contribute novel results, complementing earlier published results, of how enzyme-to-enzyme paths in metabolic pathways for *E. coli* and *S. cerevisiae* are related semantically. This is done by deriving networks where pathways are represented as nodes and edges correspond to semantic similarity-based relations. As an example, we find results indicating that there is a clear semantic relationship between paths in the sugar alcohol degradation pathways for Hexitol, Sorbitol, Mannitol and Galactitol.

5. I contribute novel results in a comparison between GO based semantic similarity and amino acid sequence similarity when assessing the ability of a path align-

ment algorithm to separate documented paths (true positives) from currently unknown paths (false positives). Results show that the two measures of similarity are complementary when deriving significant paths which are currently unknown, indicating that it is benificial to use the two measures in combination.

6. I contribute a novel method which assembles putative paths from a set of interesting genes. The paths are semantically similar to paths in documented pathway graphs. GO is used as well as known pathways, GO annotation databases, a search algorithm (e.g. an evolutionary algorithm) and a query set of gene products.

7. I contribute novel empirical results about how different search algorithms perform when assembling putative regulatory paths semantically similar to paths in documented pathway graphs. Search algorithms tested are random search, greedy search, iterated hill-climbing, stochastic hill-climbing, simulated annealing and an evolutionary algorithm (EA). Results show that greedy search is superior in terms of efficiency, but can return suboptimal solutions even for shorter paths. Therefore, iterated hill-climbing is considered to be the simplest and best alternative in terms of solution quality.

1.3 Outline of thesis

This section presents an outline of the thesis contents, and also specifies where the contributions in the previous section are justified in the thesis. Chapter 2 contains background information on relevant biology, databases, and algorithms. Chapters 3 through 5 describe the methods that were introduced in section 1.1. In general, each of these chapters contains an introduction with related work, a method description, a description of performed experiments together with results and their analysis, and a discussion section. Contribution 1 is motivated in the introduction section, and justified in the methods section of chapter 3, where the method for assessing the biological plausibility of regulatory hypotheses is described in detail. The usefulness of the method is demonstrated in the results section of the same chapter. Chapter 4 justifies contributions 2 through 5. More specifically, a detailed description of the method for performing GO-based local alignment of biological pathways is given in the methods section of this chapter, which justifies contribution 2. The contribution is motivated in the introduction section, and the utility of the method is demonstrated in the results section of the same chapter. In more detail, the results section contains empirical results showing that the richer semantic description using a combination of GO sub-ontologies results in improved sensitivity and specificity during path alignment, thereby justifying contribution 3. The same section contains the arguments for contribution 4, since experiments are described where metabolic enzyme-to-enzyme pathways for *S. cerevisiae* are compared with the same kind of pathways for *E. coli*, and semantic relations between them are established empirically. Contribution 5 is based on the arguments in the results section as well, where the alignment performance is compared for sequence- and semantic similarity. Chapter 5 justifies the last two contributions. In more detail, the arguments for contribution 6 can be found in the methods section, where the method for assembling paths similar to documented paths using GO-based semantic similarity, is described. The method is motivated in the introduction section and is demonstrated to be useful in the results section of chapter 5. The results section in the same chapter contains results from the search algorithm comparison, justifying contribution 7. Finally, conclusions drawn, ideas for

future work, and some final reflections can be found in chapter 6.

Chapter 2

Background

2.1 Molecular biology

Molecular biology involves the study of formation, function and structure of molecules such as DNA, RNA and proteins (Alberts et al. 1998). The core of the molecular biology of the cell can be described by the central dogma, which is illustrated in figure 2.1. The central dogma was presented by Francis Crick as an early draft in 1958 but in a more complete version in Crick (1970) twelve years later. The model describes how the DNA molecules are used to produce proteins. DNA (DeoxyriboNucleic Acid) is the heriditary material of an organism, and is a long polymer of small molecules called nucleotides. There are are four nucleotides; Adenine(A), Cytosine(C), Guanine(G) and Thymine(T). Nucleotides are the building blocks of genes, where where each gene is coded as a long sequence of nucleotides. An organism typically has a large number of genes. As an example, recent research shows that there are approximately 20500 human genes (Pennisi 2007). According to the central dogma, DNA can be copied to produce more DNA by replication. DNA can also be transformed into messenger RNA (RiboNucleid Acid), mRNA, during a process known as transcription. For eukaryotes, i.e. organisms with a cell nucleus, there is also a process known as alternative splicing which is performed after transcription, where sections of the mRNA are arranged into new variants. The transcription and replication takes place in the nucleus of an organism for eukaryotes. For prokaryotes (organisms lacking nucleus) these processes take place in the cytoplasm. The mRNA is subsequently transported out through the

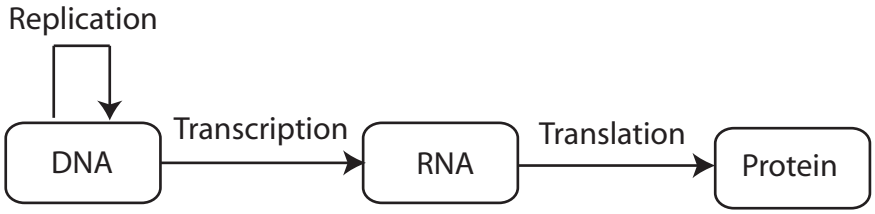


Figure 2.1: The central dogma of molecular biology, adapted from figure 7.1 in Alberts et al. (1998).

cell membrane to ribosomes in the cytoplasm where it subsequently can be used to produce proteins during the translation phase. RNA contains the same nucleotides as for DNA except for Thymine which is replaced with Uracil. It should also be mentioned that DNA is double stranded, i.e. it contains a complementary chain of nucleotides (Adenine binds to Thymine and Cytosine binds to Guanine), whereas RNA molecules are single stranded. Triplets of nucleotides along the RNA molecule, known as codons, are used to produce amino acids, which are the building blocks of proteins. A triplet codes for one of the 20 amino acids. Several different codons may code for the same amino acid, e.g. both AAU and AAC code for Asparagine. There are also special codons indicating the start and stop of a protein. As the mRNA is translated, a chain of amino acids starts to fold into a specific protein structure. There may also be special "chaperone" proteins assisting in folding the protein.

Transcription, replication and translation can be referred to as the basic processes of the central dogma. There are also less common processes, e.g. reverse transcription, where RNA is turned into DNA. This occurs for some viruses and eukaryotes under special circumstances. Another process is RNA replication which applies to some RNA viruses. It has also been shown that proteins may be translated directly from DNA (without intermediate transcription) if the process takes place in a test tube, i.e. not in a cell, and using selected cell contents of the *E. coli* bacterium.

To sum up, there is a large number of genes for each organism, and most genes have a specific task - to serve as the blueprint for the production of a specific protein. Hence, the proteins perform the actual work in terms of serving as e.g. enzymatic proteins, cell scaffolding proteins, and signalling proteins. Furthermore, genes and proteins

interact in intricate ways in order to perform different tasks in the cell. Such interaction patterns are often referred to as biological pathways, and are usually discovered by making a number of biological experiments and by studying available biological research. There are methods for measuring the amount of DNA, RNA and proteins, traditionally referred to as blotting methods, where the macromolecules are transferred to some physical carrier for analysis. A very limited number of genes/proteins can be studied at a time using such methods. An example of blotting is "Northern blot"(Alwin et al. 1977), which is used to measure the amount of RNA, and therefore how expressed a certain gene is. If a gene has a high expression, it also likely that the amount of the corresponding protein is high in the cellular tissue where the RNA was extracted. In fact, the expression of a gene results in a biochemical material known as a *gene product*, which can be either a protein or an RNA molecule, i.e. not only a protein. The advent of DNA microarray technology (Schena et al. 1995) was a revolution since it allowed the simultaneous measurement of the expression of potentially all genes in a genome. Up until this, biological experiments were rather reductionistic in nature, often studying one or a few macromolecules at a time. DNA microarrays made it possible to study how different genes interact over different experimental conditions (e.g. stimuli or time steps). Similar arrays for proteins were developed (MacBeath and Schreiber 2000), and recently also for chemical compounds (Ma and Horiuchi 2006). Microarray technology is very important to the field of systems biology, which is covered in the following section.

2.2 Systems biology

Systems biology is an emerging multi-disciplinary field in which the behaviour of complex biological systems is studied by considering the interaction of all cellular and molecular constituents rather than using a "traditional" reductionist approach, as in classic molecular biology, where constituents are studied separately in isolation (Kirschner 2005, Liu 2005). Systems are often studied over time (Aderem 2005) with the ultimate goal of developing models which can be used to predict and understand complex biological processes, such as human diseases (Butcher et al. 2004). Systems

biology also seeks to integrate information at different levels of organisation in order to understand the biology of cells. This multi-level structure can for example be illustrated by Life's complexity pyramid (Oltvai and Barabasi 2002), where there are four levels of organisation located in a pyramid structure (see figure 2.2). The broad bottom level contains organism specific entities such as genes, mRNA, proteins and metabolites. The next level is where the bottom level entities are organised into regulatory motifs and metabolic pathways. At the third level, motifs and pathways are integrated into higher order networks where tightly connected proteins and metabolites form functional modules that perform certain cellular functions. Functional modules are organised hierarchically at the top level in the pyramid in order to describe the large-scale organisation of cells. Powerful computational tools are needed at each level in this pyramid and also for integrating information between levels. In this document we propose methods for semantic comparison of different kinds of biological pathways at the second level of life's complexity pyramid. We use the Gene Ontology (GO) (Ashburner et al. 2000), documented pathways, hypothetical pathways or sets of gene products, GO annotation databases, various algorithms from computer science, and significance calculations from statistics. The use of GO makes it possible to generalise semantically about any kind of gene product. In a different track of research, not covered in this document, we have also developed different methods combining semantic and topologic analysis of protein interaction networks with the aim to identify functional modules (Lubovac et al. 2005a, Lubovac et al. 2005b, Lubovac et al. 2006a, Lubovac et al. 2006b, Lubovac et al. 2007), where GO was successfully applied in the analysis of documented Y2H (yeast-two-hybrid) networks for *S. cerevisiae*. This shows that generalising about gene products using GO is also beneficial in tools for systems biology at higher levels of life's complexity pyramid.

2.3 Graphs, networks and pathways

This section aims to explain the concepts of graphs, networks and pathways, and how they are related. Graphs are structures that are discrete and consist of vertices and

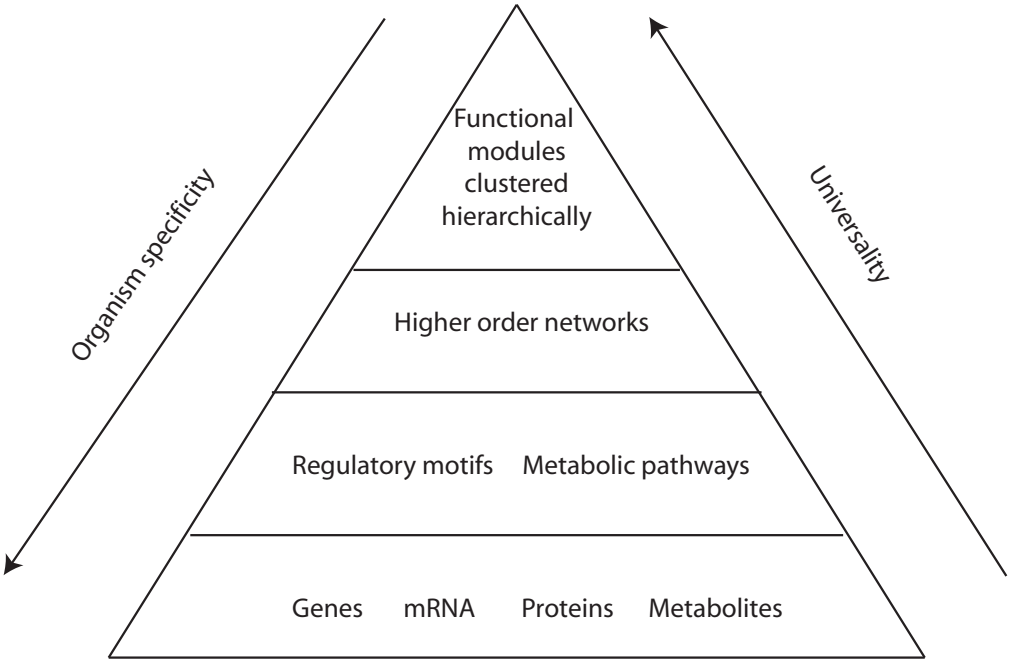


Figure 2.2: Life's complexity pyramid. Adapted from the corresponding figure in Oltvai and Barabasi (2002).

Table 2.1: Graph types. Adapted from Rosen (1995), page 433.

Type	**Edges**	**Multiple edges possible?**	**Loops possible?**
Simple graph	Undirected	No	No
Multigraph	Undirected	Yes	No
Pseudograph	Undirected	Yes	Yes
Directed graph	Directed	No	Yes
Directed multigraph	Directed	Yes	Yes

edges connecting the vertices. There are different types of graphs. A simple graph G is described mathematically as a tuple (V, E), where V is a nonempty set of vertices and E is a set of unordered pairs of distinct elements from V referred to as edges (Rosen 1995). A directed graph differs from a simple graph in that the set of edges consists of a set of *ordered* vertex pairs. One can also distinguish other variants of graphs according to Rosen (1995), which is shown in table 2.1 and illustrated in figure 2.3.

Biological networks, network motifs and pathways are all domain specific and syn-

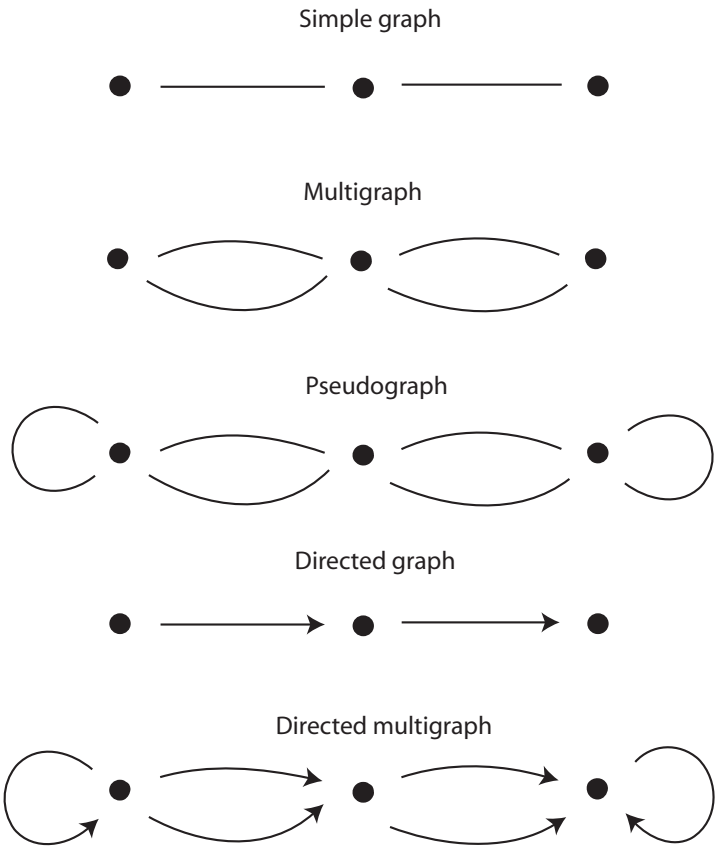


Figure 2.3: Different types of graphs.

onymous concepts which are implementations of the mathematical *graph* concept. In such graphs a vertex can for example represent a gene or a protein, and an edge can represent a regulation or other kind of interaction between two genes or proteins. Pathways are usually graphs describing biological processes taking place in a cell. There are different kinds of biological graphs. Table 2.2 shows some common examples. The methods for pathway comparison proposed in this thesis will be applied to gene regulatory networks, protein regulatory networks, and simplified (enzyme-to-enzyme) metabolic networks.

There are several important databases on biological pathways. KEGG (Kanehisa and Goto 2000) contains metabolic pathways, pathways on genetic information processing such as transcription and translation, pathways describing environmental information processing such as membrane transport and signal transduction, pathways for cellular processes describing e.g. cell motility, growth and death. KEGG also contains many pathway diagrams for human diseases such as cancers, neurodegenerative diseases and infectious diseases. There are also pathways related to drug development in KEGG. Another database collection is BioCyc (Karp et al. 2005), which contains the metabolic pathways of hundreds of organisms (MetaCyc) and also a special database for the *E. coli* metabolic pathways (EcoCyc). As another example, Biocarta (www.biocarta.com) covers a large set of pathways describing e.g. cellular processes such as cell growth and death, metabolism, immunology and neuroscience.

2.4 Gene Ontology

The Gene Ontology (GO) provides a structured vocabulary of molecular biology. The Gene Ontology consortium (www.geneontology.org) is responsible for GO, which contains three different sub-ontologies covering the molecular functions, biological processes and cellular components of gene products (Ashburner et al. 2000).

The molecular function describes activities at the molecular level that can be performed by a single gene product or complexes of gene products. Examples of activities are binding, transporter activity and catalytic activity. The gene product name is not the same as the molecular functions it can perform, and molecular function does not

Table 2.2: Examples of biological graphs. *Node* denotes what kind of molecules that are possible as nodes in a specific biological graph, and *Edge* shows the interpretation of edges.

Name	**Graph type**	**Node**	**Edge**
Gene regulatory network	Directed graph	Gene	affects transcription of
Protein regulatory network	Directed multigraph	Protein	e.g. phosphorylation, expression and methylation
Signalling network	Directed multigraph	Protein, small molecules	e.g. phosphorylation, expression and methylation
Protein interaction network	Simple graph	Protein	interacts with
Metabolic network	Directed graph	Enzyme, metabolite	catalyses, reaction

specify when, where or in what context the activity takes place. Biological process can be defined as ordered sequences of molecular functions that together accomplish some bigger task, e.g. signal transduction and cellular physiological process. A biological process is not equivalent to a pathway, even if gene products in the same pathway often have very similar or identical process annotation. Cellular component refers to a part of the cell where a gene product can be located.

The ontology is structured as a directed acyclic graph where the concepts, known as terms, are the vertices. Vertices are related to each other by relation types inheritance (IS-A) and aggregation (PART-OF), creating the edges of GO. As a consequence of GO being a directed acyclic graph, a term can have several parent terms, and a term can also have several child terms. There can be no cycles as more specific terms are connected to less specific terms and not vice versa. A small part of GO is shown in figure 2.4. There are only inheritance links in the example graph, and it illustrates the concepts of generalisation and specialisation. An increased generalisation occurs when moving from more specific terms to less specific terms, and specialisation increases when moving from less specific terms to more specific terms. The number

of GO terms is growing continuously. At time of writing (February 13, 2009), there are 8270 molecular function terms, 15963 biological process terms and 2270 cellular component terms in GO.

As an ontology is a structured and standardised vocabulary of some class of real world entities and their properties, the ontology becomes really useful when the real world entities are associated with the ontology concepts. Entities are gene products in the GO context and these are associated with GO concepts in annotation databases. For each known gene product of an organism, associations are made to the terms in GO. A gene product can be associated with several terms in each sub-ontology, because the same gene product can be involved in several different biological contexts in the cell. Currently there are GO annotations for approximately 30 different organisms available at the Gene Ontology consortium webpage, and separate annotation databases are often available for organism projects that do not collaborate with the Gene Ontology consortium. An example is *E. coli* where annotation is available at EBI (www.ebi.ac.uk).

When a gene product is annotated with a certain GO term, an evidence code is used which shows how the annotation is supported. There are a number of different evidence codes available; IC - inferred by curator, IDA - Inferred from direct assay, IEA - inferred from electronic annotation, IEP - inferred from expression pattern, IGI - inferred from genetic interaction, IMP - inferred from mutant phenotype, IPI - inferred from physical interaction, ISS - inferred from sequence of structural similarity, NAS - non-tracable author statement, RCA - reviewed computational analysis, and TAS - traceable author statement. The TAS evidence code is often believed to be the most reliable in general, where the evidence is supported by a reviewed article or book. IEA is regarded as the least reliable, where annotation is based on e.g. sequence similarity searches. However, IEA evidences may be reliable, e.g. if the e-value of a BLAST hit is very good. Hence, it can be difficult to disqualify annotations based only on their evidence codes.

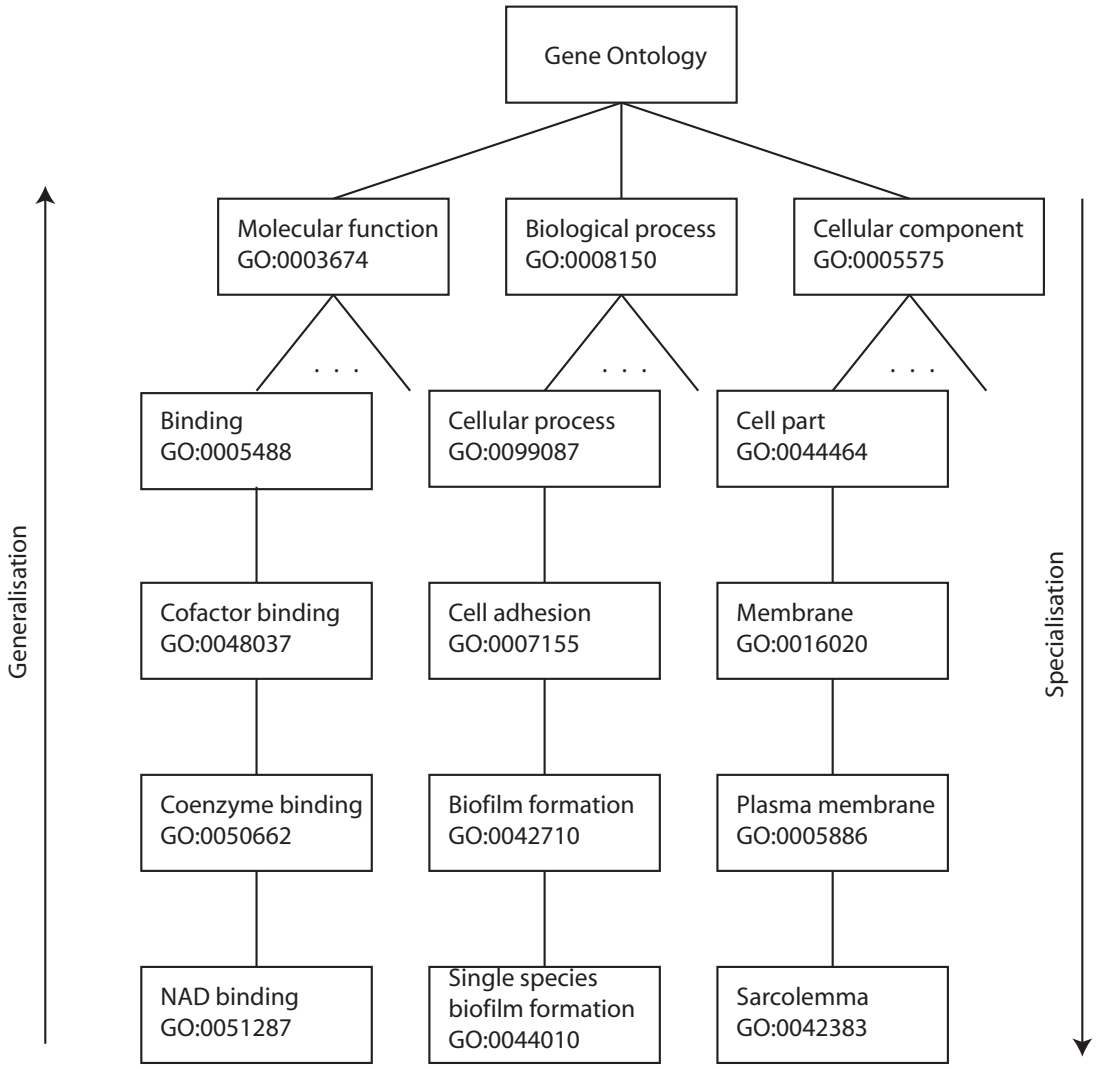


Figure 2.4: Part of Gene Ontology.

2.4.1 Semantic similarity for pathway comparison algorithms

Semantics can be defined as "the meaning or relationship of meanings of a sign or set of signs" (Merriam-Webster Online Dictionary, www.merriam-webster.com, accessed 2009-02-15). Hence, a semantic comparison of two real world entities (or signs) is about comparing the meaning of the entities. This can be contrasted to *syntax*, where a possible definition is "the way in which linguistic elements (as words) are put together to form constituents (as phrases or clauses)" (Merriam-Webster Online Dictionary, www.merriam-webster.com, accessed 2009-02-15). A syntactic comparison of two entities is in a sense less powerful as it only deals with the labels of the entities and rules concerning the validity of their labels. A very important advantage of semantic comparisons, which was also briefly mentioned in section 2.4, is that they open up for generalisation, where the meaning of real world entities is utilised by connecting entities to concepts. Generalisation requires that there are so called inheritance (or possibly aggregation) relationships between concepts documented in an ontology or abstraction hierarchy. An inheritance relation (IS-A) between two concepts means that the first concept is a specialisation of the more general second concept, and that the second concept is a generalisation of the first concept. An example is that the concept "car" is a specialisation of "vehicle", and at the same time "vehicle" is a generalisation of "car". The relation between car and vehicle can be represented as two vertices (concepts) in an ontology with an edge (inheritance relation) connecting them. A real world entity, e.g. "A Renault Laguna with registration number TNY111" can be connected to the "car" concept in this ontology. If other entities are connected to concepts in the ontology, those entities may be semantically compared with each other.

A very simple ontology containing five concepts is depicted in figure 2.5, where C1 is the most abstract term, and C4 together with C5 are most specialised. Different kinds of reasoning around semantic similarity can be performed by studying the location of the two concepts that are compared. It is for example possible to determine what concepts the entities under comparison have in common in order to get a qualitative description of why the entities are similar or dissimilar, something which is utilised by all three methods proposed in this thesis. As an example, if entity E_1 is

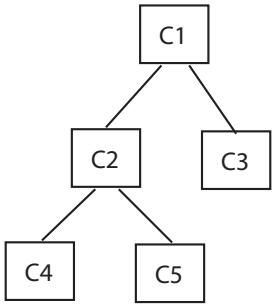


Figure 2.5: A simple ontology containing five concepts.

connected to $C4$ and another entity E_2 is connected to $C5$, we can deduce that both entities have $C2$ as the most specialised concept in common. These two entities also share $C1$ as a common concept. In this example, E_1 and E_2 are semantically similar to some degree, depending on the way semantic similarity is defined. However, if another entity E_3 that is connected to $C3$ is compared to E_1, these entities would be less similar than E_1 compared to E_2. But why is that? One (very simple but seldom used) way of measuring how similar two concepts are is to count the number of edges between them. In the first comparison there are only two edges between the concepts, and in the second comparison there are three edges. This would imply that E_1 and E_2 are more related due to the shorter edge distance. Another reason that the entities in the first comparison are more semantically similar can be that the entities share a more specialised common term. From now on (and especially in the method sections of the proposed methods in this thesis), concepts are the GO terms and entities are gene products. The fact that an entity is connected to a concept means that a gene product is annotated with a GO term in this domain of research.

A more intricate method than just counting number of edges is to perform semantic similarity calculations. For this to be possible, it is necessary to assign probabilities to the ontology terms. For GO, this can be done according to the following procedure. An annotation database D is used to calculate the probability of each GO term using the method proposed by Lord et al. (2003a). This is a well established procedure for calculation of concept probabilities in abstraction hierarchies or ontologies, and was also used by Resnik (1999) in his work on semantic similarity calculations using the WordNet (Miller 1990) vocabulary of the english language:

For each gene product G_i in D:

Increment a counter C_j for each GO term T_j appearing in the annotation of G_i, and increment the counter of each ancestor term of T_j.

For each GO term T_k:

Calculate the term probability $p(T_k) = \frac{C_k}{N}$, where N is total number of annotations in D.

Like in Lord et al. (2003a), both inheritance (is-a) and aggregation (part-of) relation types are considered in the term probability calculation. Terms of all evidence types are used, for reasons mentioned in the previous sub-section.

Different information theoretic measures have been proposed and applied to WordNet, which has semantic relations (concepts) applicable to words, e.g. nouns, verbs, and adjectives. Semantic similarity measures developed for WordNet have also been applied to the Gene Ontology vocabulary of molecular biology. The three most common measures are those proposed by Resnik (1999), Lin (1998), and Jiang and Conrath (1997). The first two are both similarity measures which compare two concepts in an ontology/abstraction hierarchy, where the Resnik measure only uses the probability of the most specific common concept and the Lin measure additionally uses the probabilities of the concepts under comparison. The measure by Jiang and Conrath is a measure of semantic distance but is otherwise similar to the Lin measure. These measures have also been used for investigating the relationship between gene sequence similarity and gene product semantic similarity (Lord et al. 2003a, Lord et al. 2003b). It was found that there is a clear correlation between these two approaches to similarity between gene products, especially when using the molecular function ontology and annotation evidence in the form of traceable author statement. Semantic similarity according to Resnik (1999) is defined as

$$SS(T_k, T_l) = -log_2(p_{ms}(T_k, T_l)) \tag{2.1}$$

where T_k and T_l are terms, $p_{ms}(T_k, T_l)$ is the probability of the minimum subsumer ms for terms T_k and T_l. The minimum subsumer ms is the ancestor term with lowest probability that terms T_k and T_l have in common. The Resnik function yields values in the interval $[0, log_2(t)]$, where t is the total number of term occurences in a corpus.

In the algorithms described in this thesis, a corpus is represented by a GO annotation file for one or several organisms. A value of 0 represents no semantic similarity at all, and $log_2(t)$ represents maximum similarity.

The measure proposed by Lin (1998) is defined in the following:

$$SS(T_k, T_l) = \frac{2log_2(p_{ms}(T_k, T_l))}{log_2(p(T_k)) + log_2(p(T_l))} \tag{2.2}$$

This measure uses both the similarity of the minimum subsumer and the similarities of the individual terms T_k and T_l. The Lin measure takes on values in the interval [0,1], where 0 represents no similarity and 1 represents maximum similarity. The latter case occurs if T_k and T_l are instances of the same term, which also means that the minimum subsumer is the same term as T_k and T_l.

A semantic distance function that is similar to the Lin measure was proposed by Jiang and Conrath (1997):

$$SS(T_k, T_l) = -2log_2(p_{ms}(T_k, T_l)) - (log_2(p(T_k)) + log_2(p(T_l))) \tag{2.3}$$

This measure varies in the interval $[0, 2log_2(t)]$, where t is the total number of term occurences in a corpus. A value of 0 represents minimum distance (maximum similarity), whereas the largest possible distance between T_k and T_l is $2log_2(t)$.

We needed to decide how appropriate the described measures of semantic similarity would be in the context of pathway comparison algorithms. In this scenario, the measure should reflect the level of specialisation of a match. This means that a match between more detailed terms with lower probabilities should also result in higher similarity. The Resnik measure exhibits this behaviour, whereas the Lin measure does not. For example, if a term is compared with itself, the Lin measure will always result in maximum similarity no matter how specialised the term is. The Jiang and Conrath measure does not have this undesirable behaviour, but is a measure of distance, and it is usually a measure of similarity that is appropriate for pathway comparison algorithms. For these reasons, the Resnik measure of semantic similarity was used for the GOSAP and GOSAM methods.

As gene products can be annotated with several GO terms, all annotations must somehow be considered. One way of doing this is by comparing terms for one gene

product with all terms for the other gene product and take the average of those similarities. By doing this, all roles of the gene product would be taken into consideration. However, when comparing pathways semantically it is of interest to obtain an explanatory model to why the pathways under comparison are similar or not. Taking the average value of all term comparisons would yield a "fuzzy" explanation to similarity since many terms are part of the explanation to some extent. An alternative approach is to take the maximum value of all term-to-term comparisons in order to derive the pair of terms that are most similar for the two gene products under study. This approach was chosen for the GOSAP and GOSAM methods in this thesis, where the semantic similarity with respect to molecular function is defined as:

$$s_f(G_i, G_j) = max(\{SS(T_k, T_l) : T_k \in t(G_i), T_l \in t(G_j)\}) \tag{2.4}$$

where G_x refers to a gene product, $t(G_x)$ is the set of GO annotations for G_x, $SS(T_k, T_l)$ is the semantic similarity between GO molecular function terms T_k and T_l, and $p_{ms}(T_k, T_l)$ is the probability of the minimum subsumer of T_k and T_l. The minimum subsumer refers to the ancestor term with lowest probability that is common to both terms. As described in the beginning of this sub-section, the probability of GO terms can be derived using annotation databases for different organisms, giving a measure of term specificity.

At path- or pathway level it is necessary to take several gene products into consideration when calculating the semantic similarity. This can in the case of paths for example be done by simply adding the similarities from each individual gene product comparison in order to create a total (combined) similarity score. As an example, if comparing the two paths $P_1 = G_{11} \rightarrow G_{12} \rightarrow G_{13}$ and $P_2 = G_{21} \rightarrow G_{22} \rightarrow G_{23}$, a combined semantic similarity could be defined as $s_f(G_{11}, G_{21}) + s_f(G_{12}, G_{22}) + s_f(G_{13}, G_{23})$. A more advanced option is to use dynamic programming approaches like the Needleman-Wunsch(Needleman and Wunsch 1970) or Smith-Waterman(Smith and Waterman 1981) algorithms, which allow paths of different lengths and present the opportunity of inserting gaps in the derived alignments. This is discussed in the following section.

2.5 Sequence alignment

This section provides background on sequence alignment since it is an important part of the GOSAP method. One of the classic and most important class of algorithms in bioinformatics are those that perform various kinds of alignments on biological sequences. According to Jacob (1977), nature is a tinkerer rather than an inventor, meaning that new sequences are adapted from already existing sequences instead of being invented from scratch. So if it is possible to detect a significant similarity between a pre-existing sequence and a new sequence, it is possible to transfer knowledge about structure and function to the new sequence (Durbin et al. 1998). Two sequences that are significantly similar are therefore likely to be *homologous* sequences, with a common evolutionary origin. However, short similar sequences may be spurious, and the similarity between sequences may be high due to the fact that both bind to the a common protein (e.g. a transcription factor). In both these cases the sequences are just similar and not homologous. It may also be the case that proteins are similar in structure, but not in sequence (remote homologs). The two most common sequence types are DNA sequences and amino acid sequences. The concept of *alignment* is important since sequences are subjected to processes of mutation during evolution such as insertions, deletions and substitutions of sequence symbols (Durbin et al. 1998). Methods for sequence alignment often originate from computer science and the analysis of text strings in general. The following subsections elaborate on different classes of sequence alignment algorithms that are used to analyse biological sequences in particular.

2.5.1 Pairwise sequence alignment

This section describes the steps required when two (biological) sequences are aligned. It is of importance to (Durbin et al. 1998); 1) adapt a scoring system for the ranking of alignments, 2) choose an alignment algorithm, and 3) choose a statistical method to assess whether an alignment score is significant or not.

1) Scoring system

In general, a total score for an alignment is calculated by summing up the similarity

between each symbol pair over the alignment (Durbin et al. 1998). Such an additive scoring scheme assumes that mutations have occured independently over the sequence. This assumption appears to be fair for DNA and amino acid sequences, but rather inaccurate for RNA sequences since there are important interactions present between residues. In order to be able to score each possible symbol pair combination it is very common to derive probabilistic substitution matrices. This is usually done by establishing a measure of the relative likelyhood that two sequences are related as opposed to being unrelated. This requires an unrelated (random) model R and a related (match) model M. The probability of two sequences being unrelated is defined as (Durbin et al. 1998):

$$P(x, y|R) = \prod_i q_{x_i} \prod_j q_{y_j} \tag{2.5}$$

where x and y are the sequences, q_{x_i} and q_{y_i} are the probabilities that symbols x_i and y_i occur independenly. The related model M is defined as:

$$P(x, y|M) = \prod_i p_{x_i y_i} \tag{2.6}$$

where $p_{x_i y_i}$ can be interpreted as the joint probability that x_i and y_i are descendants of a common (unknown) ancestor symbol. A scoring function is defined as the ratio of equations 2.6 and 2.5, also referred to as the odds ratio:

$$\frac{P(x, y|M)}{P(x, y|R)} = \frac{\prod_i p_{x_i y_i}}{\prod_i q_{x_i} \prod_i q_{y_i}} = \prod_i \frac{p_{x_i y_i}}{q_{x_i} q_{y_i}} \tag{2.7}$$

An additive scoring function known as the log-odds ratio is defined by taking the logarithm of equation 2.7:

$$S(x_i, y_i) = \sum_i log(\frac{p_{x_i y_i}}{q_{x_i} q_{y_i}}) \tag{2.8}$$

Scores calculated according to equation 2.8 can be stored in a matrix. Examples of such matrices that have been proposed are the *PAM* (Dayhoff et al. 1978) and *BLOSUM* (Henikoff and Henikoff 1992) series.

Insertions and deletions during evolution are implemented as gaps in alignments. Gaps are most often penalised in alignments. One simple way is to define a linear penalty as (Durbin et al. 1998)

$$\gamma(g) = -gd \tag{2.9}$$

or an affine gap penalty defined as

$$\gamma(g) = -d - (g-1)e \tag{2.10}$$

where g is gap length, d is the gap opening cost and e is gap extension cost. In the case of affine gap panalty, e is usually set to a smaller value than d in order to avoid excessive penalising of long gap sequences. Gap penalties may also be calculated probabilistically in a manner similar to that of substitution matrices.

2) Alignment algorithm

If the two sequences to be compared are of the same length, there can only be one global alignment involving the complete sequences, from one end to the other. But if sequences are long and of different lengths, the number of possible alignments becomes too large to enumerate. Therefore, alignment algorithms based on dynamic programming are often used, which always return optimal alignments (Durbin et al. 1998). One of the most well-known global sequence alignment algorithms based on dynamic programming is the Needleman-Wunsch algorithm (Needleman and Wunsch 1970). The first step in this algorithm is to calculate a dynamic programming matrix with accompanying traceback pointers, which is later used to derive the alignment and the alignment score. An example matrix is shown in table 2.3 where the amino acid sequences "PAWHEAE" and "HEAGAWGHEE" are aligned. The same sequences were used in an example in Durbin et al. (1998), but with a different scoring function. The size of the matrix is $(n+1) \times (m+1)$, where m and n are the lengths of the sequences.

A function F (equation 2.11) is used to calculate the matrix (Durbin et al. 1998):

$$F(i,j) = max \begin{cases} F(i-1,j-1) + s(x_i, y_j) \\ F(i-1,j) - d \\ F(i,j-1) - d \end{cases} \tag{2.11}$$

where i and j are row and column indices in the matrix, $s(x_i, y_j)$ is the scoring function, and d is the gap penalty. In the example matrix, a simple scoring function was used:

$$s(x_i, y_j) = \begin{cases} 1 & \text{if } x_i = y_j \\ 0 & \text{otherwise} \end{cases} \tag{2.12}$$

Table 2.3: Example of dynamic programming matrix for the Needleman-Wunsch algorithm when aligning the amino acid sequences "PAWHEAE" and "HEAGAWGHEE". Numbers indicate best score and arrows are traceback pointers.

		H	E	A	G	A	W	G	H	E	E
	0 ←	-1 ←	-2 ←	-3 ←	-4 ←	-5 ←	-6 ←	-7 ←	-8 ←	-9 ←	-10
	↑ ↖	↖	↖	↖	↖	↖	↖	↖	↖	↖	
P	-1	0	-1	-2	-3	-4	-5	-6	-7	-8	-9
	↑ ↖	↖	↖		↖						
A	-2	-1	0	0 ←	-1	-2 ←	-3 ←	-4 ←	-5 ←	-6 ←	-7
	↑ ↖	↖	↖	↖	↖	↖	↖				
W	-3	-2	-1	0	0	-1	-1 ←	-2 ←	-3 ←	-4 ←	-5
	↑ ↖	↖	↖	↖	↖	↖	↖	↖	↖		
H	-4	-2	-2	-1	0	0	-1	-1	-1 ←	-2 ←	-3
	↑	↑ ↖	↖	↖	↖	↖	↖	↖	↖	↖	
E	-5	-3	-1	-2	-1	0	0	-1	-1	0	-1
	↑	↑	↑ ↖	↖	↖	↖	↖	↖	↖	↖	
A	-6	-4	-2	0 ←	-1	0	0	0	-1	-1	0
	↑	↑ ↖	↖	↑ ↖	↖	↖	↖	↖	↖	↖	
E	-7	-5	-3	-1	0	-1	0	0	0	0	0

Additionally, the gap penalty d is linear and set to 1. As an initialisation $F(0,0)$ is set to 0, $F(i,0) = -id$ and $F(0,j) = -jd$. After this, the rest of the matrix is calculated starting at position $(1,1)$. This can be done either iteratively or recursively. The best scoring option in equation 2.11 is usually stored in a separate traceback matrix for each matrix position in order to be able to assemble the alignment. The value in the lower right corner, (m,n), of the dynamic programming matrix is the score of the optimal global alignment. The alignment is assembled by starting at this position and following the traceback pointers until position $(0,0)$ is reached. A diagonal traceback move $(i-1, j-1)$ in the dynamic programming matrix (see table 2.3) results in that x_i and y_j are added to the alignment. For a pointer directed to the left, $(i, j-1)$, a gap symbol "-" and y_j are added. Finally, for a pointer directed upwards, $(i-1, j)$, x_i and a gap symbol are added to the alignment. The example matrix (table 2.3) results in the following alignment:

```
---PAWHEAE
HEAGAWGHEE
```

It should be noted that although there is only one pointer originating from each matrix position in table 2.3, it is possible to implement several pointers in the case that more than one option in equation 2.11 is best. The example table only shows an arbitrary choice in such cases. Hence, there may be several global alignments that are optimal.

In contrast to global alignment, local sequence alignment algorithms search for high-scoring alignments containing fragments of the original sequences. This is often more interesting than studying alignments over complete sequences. The Smith-Waterman algorithm (Smith and Waterman 1981) is an extension to the Needleman-Wunsch algorithm. One difference compared to Needleman-Wunsch is that an additional "zero" option is added to the F function (Durbin et al. 1998):

$$F(i,j) = max \begin{cases} 0 \\ F(i-1,j-1) + s(x_i, y_j) \\ F(i-1,j) - d \\ F(i,j-1) - d \end{cases} \tag{2.13}$$

The zero option corresponds to the start of a new alignment. The application of equation 2.13 on the same sequences as before results in the dynamic programming matrix in table 2.4. Furthermore, the first column and row are initialised to 0. Another difference from Needleman-Wunsch is that the optimal alignment begins at the position with highest score rather than at the lower right corner. The final difference is that the alignment ends when the value for a matrix position is zero, i.e. if the score of the best alignment at some point is negative, it is better to start a new alignment than extending the old. Given the zero option in equation 2.13, it is assumed that the expected score of a random match is less than or equal to zero, which is the case if a log-odds ratio scoring function is used (e.g. a PAM matrix).

It turns out that there are five optimal starting positions in the matrix in table 2.4 with a score of 3. If the starting position (6,10) is chosen arbitrarily, the following alignment is obtained:

```
AW-HE
AWGHE
```

Like for the Needleman-Wunsch algorithm, several of the options in equation 2.13 may be best, yielding a collection of optimal local alignments.

In our proposed GOSAP method (see chapter 4), we apply the Smith-Waterman algorithm to the problem of aligning paths of gene products from biological pathways. However, the Needleman-Wunsch algorithm can also be applied in GOSAP if global alignments are desired.

Table 2.4: Example of dynamic programming matrix for the Smith-Waterman algorithm when aligning the amino acid sequences "PAWHEAE" and "HEAGAWGHEE".

			H		E		A		G		A		W		G		H		E		E
	0	←	0	←	0	←	0	←	0	←	0	←	0	←	0	←	0	←	0	←	0
	↑	↖		↖		↖		↖		↖		↖		↖		↖		↖		↖	
P	0		0		0		0		0		0		0		0		0		0		0
	↑	↖		↖		↖		↖		↖		↖		↖		↖		↖		↖	
A	0		0		0		1		0		1		0		0		0		0		0
	↑	↖		↖		↖		↖		↖		↖				↖		↖		↖	
W	0		0		0		0		1		0		2	←	1		0		0		0
	↑	↖		↖		↖		↖		↖			↑	↖		↖				↖	
H	0		1		0		0		0		1		1		2		2	←	1		0
	↑	↖		↖				↖		↖		↖		↖		↖		↖		↖	
E	0		0		2	←	1		0		0		1		1		2		3		2
	↑	↖			↑	↖				↖		↖		↖		↖		↖		↖	
A	0		0		1		3	←	2		1		0		1		1		2		3
	↑	↖		↖			↑	↖		↖		↖		↖		↖		↖		↖	
E	0		0		1		2		3		2		1		0		1		2		3

3) Significance test

It is possible to assess whether an alignment is biologically meaningful or not by using a significance test. In sequence alignment algorithms one alternative is using a Bayesian approach based on model comparisons. Another way is to use a classic statistical approach, where the probability that a null model results in a score greater than the score for the match, is used. The null model is that the sequences under study are unrelated. Describing these approaches in detail is beyond the scope of this thesis, see e.g. Durbin et al. (1998) for details. In our own work on GOSAP we use a different statistical significance test related to graph comparisons (see chapter 4).

2.5.2 Heuristic pairwise sequence alignment

The alignment algorithms described so far always return optimal scores given a specific scoring scheme. The Smith-Waterman algorithm is relatively slow with a computational complexity of O(nm), where n and m are the sequence lengths. There is sometimes a need for more efficient algorithms, especially when long query sequences are being processed, and large databases are to be searched for homologous matches. Heuristic alignment algorithms are approximations of optimal algorithms like the Smith-Waterman algorithm. Heuristic algorithms can usually not guarantee that the best alignment is found.

BLAST, Basic Local Alignment Search Tool (Altschul et al. 1990), is one of the

most common heuristic bioinformatics tools for local alignment of biological sequences. Abstractly, BLAST initially searches for a small seed segment from the query and database sequences where there is an exact match. Subsequently, the algorithm tries to improve the alignment score by extending the match in both directions from the seed segment. So far, no insertions or deletions are considered. If a high-scoring ungapped alignment is found, a variant of the Smith-Waterman algorithm is used in order to get a gapped alignment. A statistical significance test is also applied to the alignment in order to return a so called e-value.

2.6 Problem solving and search algorithms

Problem solving is about finding an acceptable or optimal solution to a specific problem (Michalewicz and Fogel 2004). The set of all solutions to a specific problem is often referred to as the search space. The properties of the search space are important in the choice of search algorithm, which is the means for performing the problem solving. Solutions need to be represented using a data structure suitable for the problem at hand and for many search algorithms there is a need for variation operators modifying known solutions into new solutions. Furthermore an evaluation function is useful in many cases to assess the quality of a solution. See section 2.6.1 for more details on representation, variation operators and evaluation functions.

Search algorithms either operate on complete or partial solutions. A complete solution can be evaluated at any time during the search, whereas a partial solutions gradually grow into full solutions and can therefore not be fully evaluated until the end of the search. Algorithms with stochastic elements often operate on complete solutions, e.g. evolutionary algorithms, simulated annealing and hill-climbing. Examples of algorithms operating on partial solutions are different kinds of tree search algorithms such as depth first search, greedy search, A*, and branch & bound, where solutions are assembled by visiting the nodes in a tree according to a certain strategy. Greedy search algorithms can also be applied to other solution representations than trees.

When searching for solutions in a search space, it is essential to get a good balance

in terms of *exploration* and *exploitation* (Michalewicz and Fogel 2004). Exploration is about moving from a current solution into unknown territories, often by introducing noise to the solution using a mutation operator. The exploitation part is about using known parts of the search space to create new solutions, this can be represented e.g. by a crossover operator in an evolutionary algorithm (EA) or a strategy for choosing the best next solution in a hill-climber or greedy algorithm. A good search algorithm needs to do both parts. An algorithm that only explores new territory will not record any good solutions in the vicinity of the current solution, and will merely perform a random walk in the search space. By only exploiting the currently known search space, there will be no chance of finding other promising parts of the search space, and will most often lead to suboptimal solutions being stuck on local optima. The multi-armed bandit model introduced by Robbins (1952) is illustrating the balance between exploration and exploitation. This is a problem featuring a gambling machine similar to a one-armed bandit but with multiple levers. The player must maximise the profit over a finite number of lever pulls. However, the player has no prior knowledge of the distribution of rewards from the different levers. The only way to acquire knowledge of the distribution is to perform multiple pulls on each lever. The problem is to know when to continue to play on a specific lever (exploitation) and when to move on to try other levers (exploration).

Some search spaces are multi-dimensional and unpredictable, and may therefore require more sophisticated search algorithms than other search spaces. The no free lunch theorems proposed by Wolpert and Macready (1997) are important in order to understand the implications of the choice of search algorithm. The theorems state that if an algorithm performs very well on one certain class of problems then it must perform worse on average on all other classes of problems. This implies that a particular algorithm with its set of parameter settings in general has limited applicability, and that every algorithm performs equally well on the set of all possible problems. Hence, if little is known about the problem and its search space, it is not obvious what search algorithm to use. For this reason, several distinctly different search algorithms are evaluated in the GOSAM method in chapter 5.

In e.g. the area of inductive learning and classification there is a design principle

often used which is known as Ockham's razor, which states that the most likely hypothesis is the simplest hypothesis consistent with the set of observations used (Russell and Norvig 1995). This can for example correspond to the decision tree with fewest decision variables if there are bigger trees that equally well represent the set of observations. In many cases a smaller classifier has better generalisation performance than a larger classifier when classifying future, yet unseen, observations. In the area of problem solving when searching for solutions, Ockham's razor could suggest that the simplest search algorithm with a small number of parameters would be the one to prefer if more complex algorithms with more parameters only offer the same observed performance for the problem under study. We shall use this principle in the evaluation of different search algorithms intended for the GOSAM method in chapter 5.

2.6.1 Evolutionary algorithms

Evolutionary algorithms constitute a family of algorithms that can be used for optimisation and general problem solving, and are introduced here because they are useful in the context of GOSAM. A population of potential solutions to a problem is maintained and improved over time by the use of various operators for selection and genetic variation. The generic evolutionary algorithm is shown in pseudo code in figure 2.6 (Michalewicz and Fogel 2004). A population of n candidate solutions $P(t) = \{s_1^t, \ldots, s_n^t\}$ for iteration t is maintained by the algorithm and is initialised in the beginning. The initial solutions may be randomised or configured arbitrarily. The set of solutions is evaluated and a new population of n solutions is selected from the current population at the next iteration. The new population is subjected to a set of genetic variation operators and evaluated. At the next iteration a new population is selected, and so on. This procedure continues until some termination criterion is reached, which could be that a predetermined number of iterations have elapsed or that the quality of the solutions have reached a sufficiently high level at the evaluation. A more detailed description of the various aspects of evolutionary algorithms is given in the following.

A *solution representation* is a mapping from the state space of possible solutions to a state space of encoded solutions using a particular data structure (Michalewicz

```
t ← 0
initialise P(t)
evaluate P(t)
while(not(termination-condition)) {
    t ← t + 1
    select P(t) from P(t − 1)
    alter P(t)
    evaluate P(t)
}
```

Figure 2.6: The generic evolutionary algorithm.

and Fogel 2004). Example representations are bit strings, floating point or integer vectors, matrices, and graphs. There are even more advanced representations. A distinction can be made between representation at the phenotype- and genotype level. A solution contains a set of parameters which describe the solution at the phenotype level. A chromosome contains genes which describe the solution at the genotype level. The genotype level is used when genetic variation operators are applied whereas the phenotype level describes the solution as used during fitness evaluation. A mapping function is used for transformation from the genotype level to the phenotype level. Hence, a gene encodes the value of a parameter. It is usually a good idea to choose a representation that suits the problem to be solved. The solution representation and variation operators are closely related in an evolutionary algorithm and the choice of representation affects the choice of variation operators.

An *evaluation function* is applied to the set of solutions in order to assign a fitness value to each solution (Michalewicz and Fogel 2004). A fitness value reflects how good each solution is according to a specified evaluation function. The evaluation function is the interface to the actual problem and constitutes a problem specific implementation of an evolutionary algorithm. It is most common to maximise the fitness, i.e. good solutions are assigned high fitness values. Another possibility is fitness minimisation, but some selection operators may be incompatible with this approach.

The *selection operator* determines how a new population is selected from the current population (Michalewicz and Fogel 2004). Selection is guided by the fitness values assigned to the solutions in the population. Selection methods can be classified as either deterministic and stochastic. Deterministic methods always select the same solutions for a specified population whereas stochastic methods are probabilistic. Selection is about choosing parents for a new population but also about choosing what solutions to replace in each generation. A possible distinction can be made between generational- and steady state evolutionary algorithms. A generational evolutionary algorithm replaces the entire population at each iteration step whereas a steady state evolutionary algorithm replaces only a subset of the population.

A common addition to an evolutionary algorithm is the *elitist strategy* where the best solution(s) are transferred to the next generation in order to maintain a monotonous growth in fitness. This strategy usually improves the performance of the algorithm considerably. An elitist strategy is an example of a *steady state* extension to an EA. On the contrary, in *generational* EAs there is no guarantee of monotonous fitness growth, because solutions are not guaranteed to survive to the next generation.

A subset of the selected solutions are subjected to a set of *genetic variation operators* with the purpose of adding new information to the solutions (Michalewicz and Fogel 2004). Variation operators are designed to fit the representation that is used for a particular problem. Variation operators perform unary or higher-order transformations. Unary transformations are performed when new solutions are generated by altering a single individual. Mutation operators usually perform unary transformations, e.g. by flipping a bit in a bit string or adding noise to a parameter in a floating point vector. Higher-order transformations generate new solutions by combining parts from several solutions. Crossover operators perform higher-order transformations where parts of several solutions are swapped at one or several split points.

There are a number of advantages of evolutionary algorithms (Michalewicz and Fogel 2004): (1) EAs are conceptually simple and relatively easy to implement, (2) EAs can be applied to a vast number of problem domains, (3) EAs can relatively easily be extended to hybrid search algorithms, (4) EAs are highly parallel and can operate

efficiently in multiprocessor environments, (5) EAs are robust to dynamic changes in the problem environment, (6) EAs can be organised to employ self-adaptation of control parameters, (7) EAs can perform problem solving in domains where there are no known solutions, e.g. in large search spaces where solutions are hard or impossible to formalise.

A steady state EA is for these reasons tested within the GOSAM method in chapter 5, and this EA has as representation a permutation of gene product symbols.

2.6.2 Non-evolutionary search algorithms

A question that should be asked is whether there are other, non-evolutionary, optimisation algorithms that perform better than an EA in GOSAM. Even if EAs have been successful in the search for solutions to many problems, there are other alternatives available which may be very well suited to this particular problem. Simple algorithms like hill-climbing and greedy search are often shown to have a more suitable speed/quality tradeoff especially when applied to larger search spaces. In order to investigate this, five different algorithms were tested. The algorithms tested and described in the following, are: random search, greedy search, iterated hill-climbing, stochastic hill-climbing and simulated annealing.

Random search

Random search simply generates and evaluates a fixed number of random permutations from a query set of gene products, and keeps track of the best solution. Hence, random search explores solutions in the entire search space in a very unorganised manner. The pseudo-code is shown in figure 2.7. In our simulations N was set to $1.5 \cdot 10^5$.

Greedy search

A greedy search algorithm typically constructs a complete solution iteratively. At each iteration the current solution is extended using the most favourable option at the moment (Michalewicz and Fogel 2004). A greedy algorithm does not guarantee an optimal complete solution. The greedy algorithm we developed for deriving paths is

```
Select current solution v_c at random
v_best ← v_c
i ← 1
while(i < N) {
    Select current solution v_c at random
    if(eval(v_c)>eval(v_best))
        v_best ← v_c
    i ← i + 1
}
```

Figure 2.7: The random search algorithm.

shown in figure 2.8, where M is the set of gene products in the model, Q is the query set, and $SS_{GP}(M_i, Q_j)$ is the composite semantic similarity between gene products M_i and Q_j (see equation 5.6). The algorithm performs one greedy search and can be restarted several times in order to increase the chance of finding good solutions.

Iterated hill-climbing

Iterated hill-climbing is a local search algorithm which starts from a random solution and explores if there are any better solutions in a neighbourhood of that solution. This is done by applying the mutation function to the current solution. If a neighbourhood solution is better than the current solution, a new neighbourhood exploration will be performed from the new solution. This continues until no improvement is possible within a maximum number of neighbourhood trials. In that case, the algorithm is restarted with a new random solution. The implemented algorithm (see figure 2.9) is based on the one described in figure 5.1 in Michalewicz and Fogel (2004), but with one important difference: The algorithm proposed in Michalewicz and Fogel (2004) suggests that *all* solutions in the neighbourhood are explored and that the best of these solutions is selected. However, as there typically are so many possible neighbours to a solution for the problem under study, a limited number of mutated solutions are tested. The first better solution is accepted rather than the best of all

```
Shuffle Q randomly
i ← 0
while(i < size(M)) {
    Score_b ← SS_GP(M_i, Q_0)
    Pos_b ← 0
    j ← 1
    while(j < size(Q)) {
        Score ← SS_GP(M_i, Q_j)
        if(Score > Score_b) {
            Score_b ← Score
            Pos_b ← j
        }
        j ← j + 1
    }
    Remove Q_{Pos_b} from Q
    Add Q_{Pos_b} and M_i to alignment
}
```

Figure 2.8: The greedy search algorithm.

```
i ← 0
while(i < N) {
    Select current solution v_c at random
    if(i = 0)
        v_best ← v_c
    local ← TRUE
    while (local) {
        j ← 0
        local ← FALSE
        while (j < M and not(local)) {
            Mutate v_c into v_n
            if(eval(v_n)>eval(v_c) {
                v_c ← v_n
                local ← TRUE
                if(eval(v_n)>eval(v_best))
                    v_best ← v_n
            }
            j ← j + 1
        }
    }
    i ← i + 1
}
```

Figure 2.9: The iterated hill-climbing algorithm, based on the algorithm in figure 5.1 in Michalewicz and Fogel (2004).

neighbour solutions. Hence, our algorithm is a less stringent local search as it may not necessarily make a greedy move towards the best solution. The speed of the algorithm also improves when not searching for the best solution in the neighbourhood. Local optimization algorithms like hill-climbing are often fast, but the main drawback is that they tend to get stuck in local optima.

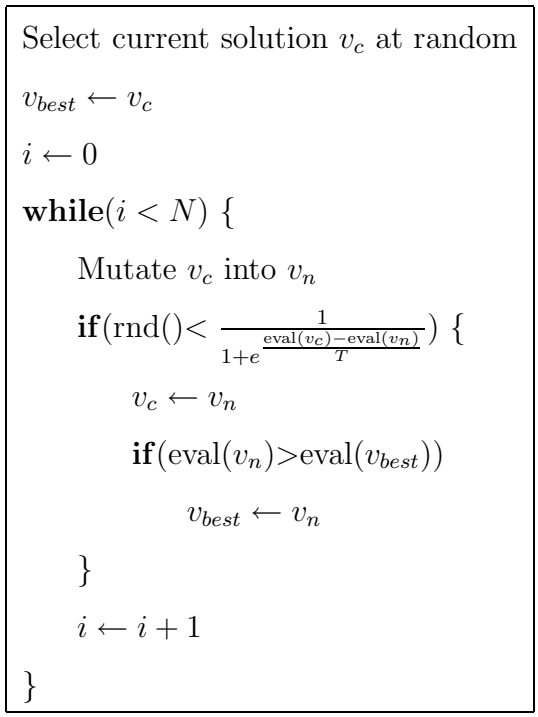


Figure 2.10: The stochastic hill-climbing algorithm, based on the algorithm in figure 5.2 in Michalewicz and Fogel (2004).

Stochastic hill-climbing

The stochastic hill-climbing algorithm, see figure 2.10, is different from iterated hill-climbing in that it accepts a new solution in the neighbourhood with a certain probability which depends on the difference in evaluation when comparing with the current solution and the value of a constant T. Hence, stochastic hill-climbing tries to escape local optima by probabilistically allowing bad moves, which increases the chance of finding the global solution to a problem rather than a local (non-optimal) solution. The T constant needs to be set appropriately, taking the evaluation function into consideration. Table 2.5 shows examples of probabilities of accepting a new solution given different T-values. It is evident that a small T makes the algorithm less prone to accept new solutions that are worse than the current solution, i.e. the algorithm will operate like an ordinary hill-climber. Larger T increases the probability of accepting "bad" moves. Therefore, it is important to know how big differences in evaluation that are expected in order to choose a suitable T.

Table 2.5: Probability of accepting new solutions as a function of eval(v_n) and T when eval(v_c)=0.85

eval(v_n)	T	eval(v_c)-eval(v_n)	$e^{\frac{\text{eval}(v_c)-\text{eval}(v_n)}{T}}$	$p = \frac{1}{1+e^{\frac{\text{eval}(v_c)-\text{eval}(v_n)}{T}}}$
0.8000	0.1000	0.0500	1.6487	0.3775
0.8490	0.1000	0.0010	1.0101	0.4975
0.8500	0.1000	0	1.0000	0.5000
0.8510	0.1000	-0.0010	0.9900	0.5025
0.9000	0.1000	-0.0500	0.6065	0.6225
0.8000	0.0100	0.0500	148.4132	0.0067
0.8490	0.0100	0.0010	1.1052	0.4750
0.8500	0.0100	0	1.0000	0.5000
0.8510	0.0100	-0.0010	0.9048	0.5250
0.9000	0.0100	-0.0500	0.0067	0.9933
0.8000	0.0010	0.0500	$5.1847 \cdot 10^{21}$	0.0000
0.8490	0.0010	0.0010	2.7183	0.2689
0.8500	0.0010	0	1.0000	0.5000
0.8510	0.0010	-0.0010	0.3679	0.7311
0.9000	0.0010	-0.0500	0.0000	1.0000

Simulated annealing

The main difference between simulated annealing (see figure 2.11) and stochastic hill-climbing is that T (also referred to as temperature) changes over time. Additionally, better solutions are always accepted whereas worse solutions are accepted probabilistically. The idea is to explore large parts of the search space in the beginning of the search when T is large, and to reduce T over time to obtain a hill-climbing behaviour at the end of the search. By doing this, the risk of finding sub-optimal local solutions decreases and the chance of finding global solutions increases (Michalewicz and Fogel 2004). However, as optimisation algorithms get more intricate and powerful, the number of parameters usually increases. Parameters for simulated annealing are start temperature T, final temperature T_{min}, number of inner loop iterations N and temperature decrease ratio r. Setting these parameters in an optimal manner can be difficult. The minimum temperature can be calculated as $T_{min} = Tr^i$ where i is number of temperature iterations.

```
Select current solution v_c at random
v_best ← v_c
while(T < T_min) {
    i ← 0
    while(i < N) {
        Mutate v_c into v_n
        accept ← FALSE
        if(eval(v_n)>eval(v_c))
            accept ← TRUE
        else if(rnd()< e^((eval(v_n)−eval(v_c))/T))
            accept ← TRUE
        if(accept) {
            v_c ← v_n
            if(eval(v_n)>eval(v_best))
                v_best ← v_n
        }
        i ← i + 1
    }
    T ← rT
}
```

Figure 2.11: The simulated annealing algorithm, based on the algorithm in figure 5.3 in Michalewicz and Fogel (2004).

Chapter 3

GOTEM: GO-based regulatory TEMplates

Many algorithms have been proposed for deriving regulatory networks from microarray gene expression data. The performance of such algorithms is often measured by how well the resulting network can recreate the gene expression data that it was derived from. However, this kind of performance does not necessarily mean that the regulatory hypotheses in the network are biologically plausible. We therefore propose a method for assessing the biological plausibility of regulatory hypotheses using prior knowledge in the form of regulatory pathway databases and Gene Ontology-based annotation of gene products. A set of templates is derived by generalising from known interactions to typical properties of interacting gene product pairs. By searching for matches in this set of templates, the plausibility of regulatory hypotheses can be assessed. We evaluate to what degree the collection of templates can separate true from false positive interactions, and we illustrate the practical use of the method by applying it to an example network reconstruction problem.

3.1 Introduction

It is highly desirable to be able to derive causal gene regulatory networks using gene expression data. This is known as reverse engineering of genetic networks (D'haeseleer et al. 2000). Time series expression data is often used, and the reverse engineering algorithm tries to find a set of activation rules that fits the data. The rule set should generate the expression levels for the genes in the current time step using the expression levels of the previous time step. Methods for reverse engineering of genetic networks can use boolean networks (Liang et al. 1998, Akutsu et al. 1999) where expression levels are discretised as on or off, or continuous additive neural network inspired models (D'haeseleer et al. 1999, Weaver et al. 1999) which use the actual expression levels. Furthermore, Bayesian networks have been proposed by a number of researchers (Friedman et al. 2000, Kim et al. 2003, Husmeier 2003, Imoto et al. 2003, Zhou et al. 2004, Zou and Conzen 2005, Geier et al. 2007).

In most cases, many different reverse engineered networks are consistent with the observed data, but we can expect that only some of these networks are biologically plausible. A drawback with methods for network reconstruction which are based solely on fit to the data is that they do not provide any way of distinguishing between biologically plausible and implausible networks. It has been proposed to include previous knowledge by including known regulatory interactions in the network reconstruction process, but this approach suffers a lack of generality: already known relations would be rediscovered, while no help is provided for discovering relations that are functionally similar, but previously unknown. This limitation seems crucial, since the objective of the whole exercise is to discover previously unknown relations.

Therefore our aim is to propose a method for assessing the biological plausibility of regulatory hypotheses. The method should be able to utilise general knowledge about regulation in known relations in the assessment of new hypothetic relations derived by e.g. reverse engineering algorithms.

The mismatch between plausibility with respect to the data and biological plausibility has been observed before. It has been claimed that gene microarray data alone is not sufficient for accurate regulatory network derivation (D'haeseleer et al. 2000).

Hence, some work on genetic networks use different types of prior knowledge in the inference process. Hartemink et al. (2002) used a combination of Bayesian networks, temporal gene expression data and data from analysis of genomic binding locations in the inference of the the pheromone response pathway in *S. cerevisiae*. However, this approach uses specific knowledge in the network inference, rather than general knowledge as we propose.

A useful resource to support our ideas of general knowledge is the Gene Ontology (GO), which was introduced earlier and provides means for generalising about gene products and their semantic properties. Apart from using GO to achieve generalisation, the concept of templates is appealing for identifying hypothetic regulatory relations that are similar to known regulatory relations. Templates in the context of association rule discovery and expression data was used by Tuzhilin and Adomavicius (2002). A template-language was designed that allowed a user to group, filter and inspect the large number of rules produced by association rule discovery algorithms. Of particular interest for our work are rule templates $F1 \rightarrow F2$ which detect rules where the antecedent part contains any of the genes in a predefined functional group $F1$ and the consequent part contains a gene from a group $F2$. The functional groups of genes were defined manually by an expert user. It should be noted that association rules are not causal relations; an association rule $A \rightarrow B$ merely claims that object A co-occurs with B.

As an example of using GO and semantic similarity as prior knowledge, Speer et al. (2004) combined correlation distances between expression profiles of genes and semantic distances between the biological process terms of the corresponding gene products in a memetic co-clustering algorithm. It was found that the GO annotations are useful for finding clusters of genes participating in similar biological processes. GO has also been used for informed regulatory network inference by combining gene expression data, GO process terms and transcription factor binding site information (Haverty et al. 2004). However, in this work GO was not used to generalise about gene products and their semantic properties.

Our contribution is GOTEM, a Gene Ontology based method for assessing the biological plausibility of regulatory hypotheses at the gene product level using prior

biological knowledge in the form of Gene Ontology annotation of gene products and regulatory pathway databases. The templates are designed to encode general knowledge, derived by generalising from known interactions to typical properties of interacting gene product pairs. By matching regulatory hypotheses to templates, plausible hypotheses can be separated from implausible hypotheses. GOTEM is published in Gamalielsson et al. (2006), and was prior to this filed as a technical report (Gamalielsson et al. 2005).

Subsequently to our publication, the idea of using functional GO pathway templates was proposed by Cakmak and Ozsoyoglu (2007) in order to discover unknown pathways in newly sequenced organisms, but this purpose is different from the purpose of GOTEM.

3.1.1 Related work

Since the approach to rule templates in the context of association rule discovery applied to gene expression data (Tuzhilin and Adomavicius 2002, Adomavicius 2002) is the main inspiration to our idea of regulatory templates, a comprehensive description of their work is given. As a background, association rule discovery algorithms aim to find relationships between items in a database. An example of an implementation is the Apriori algorithm (Agrawal and Srikant 1994). A database is obtained by collecting data from the domain under study, e.g. different expression levels of the genes for a certain condition in a microarray experiment. Being more specific, the database contains a set T of transactions, where each transaction t_i contains a set I_i of items. An association rule is an if-then construction on the form $LHS \rightarrow RHS$, where RHS and LHS are disjunct sets of items. The semantics of an association rule are: if the items in LHS occur, the items in RHS will also occur. Since a huge amount of rules are derived using association rule discovery, it is of interest to filter out the most interesting ones. The two most common measures of objective rule interestingness are *support* and *confidence*. Support is the proportion of transactions that are covered jointly by LHS and RHS, whereas confidence is the proportion of transactions covered by LHS that are also covered by RHS. However, there is a need for more advanced methods for filtering and validating association rules. Adomavi-

cius (2002) proposes a method for expert-driven validation of set-based data mining results, such as rules induced by association rule discovery algorithms. Abstractly described, the method relies on a domain expert's iterative use of various validation operators that can handle multiple patterns (e.g. association rules) at a time. In an application paper using the method developed in Adomavicius (2002), Tuzhilin and Adomavicius (2002) address the problem of post-processing the large amount of rules generated by association rule discovery algorithms in the context of microarray gene expression data. They propose a set of tools to be used by a biologist for post-processing purposes including rule filtering, grouping, browsing and data inspection. The idea is that the biologist can use these tools iteratively and interactively to make biological discoveries. The approach described assumes that gene expression levels are discretised as either upregulated ($\uparrow$), downregulated ($\downarrow$) or unchanged (#). Each transaction in the database corresponds to one experiment with the expression levels of all the genes on the microarray. The rule filtering is implemented with a template language, where templates have the form "*Rulepart* HAS $Quantifier$ OF $C_1, C_2, \ldots, C_N[ONLY]$". *Rulepart* can be BODY, HEAD or RULE, and specifies what part of the rule the filter applies to. $C_1, C_2, \ldots, C_N$ is the comparison set containing a specification of what genes (together with possible gene expression levels) that the discovered rules will be compared with. $Quantifier$ specifies how many genes that must be included in the *Rulepart* of the discovered rules. The alternatives are (ALL), (ANY), (NONE) or a list of exactly what genes to include. The optional $ONLY$ parameter specifies that no other genes than those stated in $C_1, C_2, \ldots, C_N$ are allowed. Several rule templates can be combined creating a composite filter using the keywords AND, OR, and NOT. Additionally, the macro templates POSSIBLE_INFLUENCE and CONTRADICT provide extended filtering possibilities and are composite filters. POSSIBLE_INFLUENCE(GeneSet1, GeneSet2) returns those rules that has some genes in GeneSet1 in the antecedent and some genes in GeneSet2 in the consequent, or vice versa. The CONTRADICT macro implements the idea of unexpectedness of rules according to work such as Liu and Hsu (1996) and Padmanabhan and Tuzhilin (1999). By using CONTRADICT(GeneExprSet, G, Explevel), it is possible to find rules which contradict the hypothesis that a set of gene expres-

sions GeneExprset induces gene G with expression Explevel. An example could be CONTRADICT({G1↑, G2↑}, G4, ↓), where any rule containing G1 and G2 upregulated and G4 upregulated or unchanged, are detected as unexpected. Many rules are similar among the large amount of rules discovered, and it is desirable to be able to study similar rules together. This is because the sheer amount of rules is reduced and that a comprehensible overview of the rules is obtained. Rule grouping in Tuzhilin and Adomavicius (2002) is based on the concept of gene hierarchies. Each gene is assigned to a functional group (e.g. DNA repair) by a biologist and this procedure creates a hierarchy of functional groups. In this way aggregated rules are created. For example, two functional groups F1={G1,G2,G3} and F2={G4,G5} could generate an aggregate rule R=F1→F2. Simple rules complying with the aggregate rule can only contain genes from F1 in the antecedent part and genes from F2 in the consequent part. In this case the aggregate rule would include the simple rules R1=G1↓→G4↓, R2=G1↓→G5↓ and R3=G1↑ & G3↓→G5↓. Additionally, aggregated rules with expression levels are possible. The rule R'=F1↓→F2↓ would include rules R1 and R2, but not R3. In addition to filtering and grouping of rules, there are operators where the biologist can select and view certain rules or groups of rules and explore these in a more detailed manner. Functionality for displaying what experiments (transactions) contribute to a rule was also included. The toolbox of rule exploration operators was tested on a data set showing gene expression in the yeast *Saccharomyces cerevisiae* in response to various treatments, and the resulting data matrix contained 28 experiments and 2600 genes. Different biological questions were posed, e.g. "When genes involved in the DNA repair are upregulated, what other gene categories are also up- or downregulated?". The biological questions were transformed into applications of rule exploration operators. In this particular case the number of rules were reduced from 70 millons to 1673 using the rule template "BODY HAS (ANY) OF [DNA_Repair]↑ ONLY AND HEAD HAS (ANY) OF [All_Genes]={↑,↓}". The genes were also grouped according to their primary functional category, which in this particular case resulted in a reduction from 1673 simple rules to 78 aggregated rules of direct interest to biologists. No use of the unexpectedness operator CONTRADICT was reported in this context. However, several questions of interest to biologists could

be answered by using the approach to rule exploration proposed in the paper.

The idea of regulatory templates in terms of network motifs is described in Lee et al. (2002), where it was determined how the 141 known transcriptional regulators of *S. cerevisiae* associate with all genes in the genome. A technique called genome wide location analysis was used to do this by measuring how the regulating protein binds to promoter regions over the entire genome. Using a strict statistical significance policy in the method, approximately 4000 interactions were detected. 2343 of all 6270 genes in the genome had at least one regulator associated. Each regulator binds to 38 promoter regions on average. Network motifs are defined as small graph templates of regulation, each with certain "topology rules". The location data was used to derive six different types of commonly occuring network motifs: 1) the autoregulation motif, where a regulator binds to the promoter region of its gene, 2) the multicomponent loop motif, which is a regulatory circuit (loop) containing at least two regulators, 3) the feedforward loop motif, containing a regulator which controls another regulator, and both regulators bind to the same target gene, 4) the single input motif, where one regulator binds to a set of genes, 5) the multi-input motif, where a set of regulators bind to the same set of genes, and 6) the regulator chain motif, which is a chain of at least three regulators where each regulator binds to the promoter region of the next regulator. Using a refined motif derivation procedure involving microarray gene expression data from hundreds of experiments, it was possible to derive (mostly multi-input) motifs that can be used to reconstruct the network structure for the *S. cerevisiae* cell cycle without prior knowledge of the involved regulators. The authors claim that the approach has general applicability for deriving regulatory network structures in higher eukaryotes as well.

Another paper similar to Lee et al. (2002) also addresses network motifs. The *E. coli* transcriptional regulatory network was mined for frequently recurring network motifs in Shen-Orr et al. (2002). Three common patterns were found: 1) the feed forward loop, which is similar to the one in Lee et al. (2002) but the transcription factors regulate an operon (group of contiguous genes that are transcribed into one mRNA molecule), 2) Single input module, where one transcription factor regulates a set of operons, and 3) dense overlapping operons, where a set of operons are regulated

by combination from a set of transcription factors. Results also indicate that motifs possibly have specific functions in the information processing taking place in the network. As an example, the feed forward loop is often found where a rapid response in a system is caused by an external signal. Additionally, the motifs can be used to represent the full regulatory network in a compact and modular form.

Cakmak and Ozsoyoglu (2007) propose a method for discovering unknown pathways in newly sequenced organisms. The idea is to use known metabolic enzyme-to-enzyme pathways for a number of species, and convert those to what the authors refer to as "Pathway Functionality Templates" (PFT). A PFT has the same topology as the original pathway, but the enzymes are replaced by the most specific molecular function term in Gene Ontology. Subsequently, all organism specific versions of a certain pathway are scanned for frequently occuring PFT sub-graphs, so called PF patterns. The functional metabolic network of an organism where the individual metabolic pathways are unknown, is then searched for matches to the frequent PF patterns. A database of 30 bacterial organisms, each with the same 50 organism-specific metabolic pathways, was used. To assess the performance, a leave-one-out cross validation was performed. One at time, each of 30 organisms was excluded from the mining of frequent PF patterns. Using the measures precision and recall, the share of correctly predicted PF patterns was evaluted. The results show that nodes in the pathways were predicted with 86% precision and 72% recall, and interaction predictions had a precision of 85% and 64% recall. Apart from the leave-one-out test of the method, the authors show that the method is able to derive novel metabolic enzyme-to-enzyme pathways for *S. cerevisiae*. The correctness of the edges was assessed by checking if there is a high correlation between gene expression profiles for the two genes connected to an edge. For a majority of the edges, there is a clear correlation. The authors only present results for metabolic pathways, but claim that the method can be modified easily to cover e.g. signalling pathways. It is also noted that the most specific GO term for an enzyme could be replaced by an ancestor term in order to allow for further generalisation, but it is not automatically supported in the current version of the method, motivated by that this would result in a large amount of false positive matches. However, the method detects a match if the GO term to be

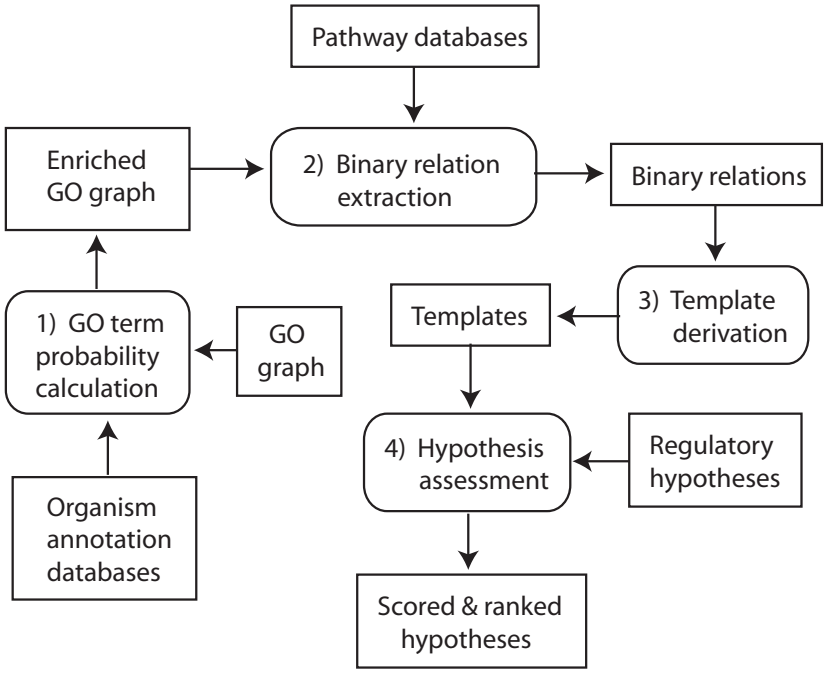


Figure 3.1: The GOTEM method. Boxes with rounded corners represent procedures, and rectangular boxes represent information.

matched is more specific than the one in the PF pattern. Authors also mention the idea of using semantic similarity measures for PF pattern matching.

3.2 Method

The basic idea is to use what we know about regulation in documented pathways, generalise this knowledge, and apply the generalised knowledge when assessing the biological plausibility of hypothetical regulatory relations. The GO molecular function classification of the gene products participating in regulatory relations in pathways is used to derive templates. The templates encode the types of gene products known to be involved in a particular type of regulatory relation.

More specifically, the method can be divided into four major procedures that are illustrated in figure 3.1 and described in detail in the following subsections: (1) GO term probability calculation, (2) binary relation extraction, (3) template derivation and (4) Hypothesis assessment.

3.2.1 GO term probability calculation

The aim of the first step is to calculate how specific different GO terms are by using an annotation database for all the gene products of a genome. The GO term probability calculation is performed according to the procedure described in Lord et al. (2003a) and the annotation databases available in september 2004 at the Gene Ontology website (http://www.geneontology.org) for *S. cerevisiae* (SGD) and *H. Sapiens* (EBI), are used. Additionally, annotations with all types of evidence are used. TAS is regarded to be the most reliable type of evidence, but only using TAS would result in losing a very large proportion of the gene products, especially for *H. Sapiens* (see table 3.1). Therefore we chose to use all annotations, despite the risk of introducing less reliable evidence.

Table 3.1 shows the GO annotation evidence distribution for the gene products in the cell cycle regulatory pathway for *S. cerevisiae* and *H. Sapiens* according to KEGG (http://www.genome.jp/kegg). The distribution for the whole genome is also shown. Only gene products with known functional GO annotation are used (i.e. not "GO:0005554", the code for unknown function). It can be noted that 53.2% of the gene products in the *S. cerevisiae* model are annotated with traceable author statement, compared to 26.3% for the entire genome. For *H. Sapiens* the proportion of TAS entries is only half of that for *S. cerevisiae*. Furthermore, evidence inferred from electronic annotation dominates for *H. Sapiens* with 53.9% for the cell cycle pathway, because homology to model organisms is often used in the annotation. The term probability calculation is explained in the following procedure where A-db is a GO annotation database for gene products (e.g. SGD):

For each gene product GP_i in A-db:

Increment a counter CNT_j for each GO term GT_j in the annotation of GP_i and increment the counter of each ascendant term of GT_j.

For each GO term GT_j:

Divide CNT_j with the total number of GO term annotations in A-db, yielding the term probability $p(j) \in [0, 1]$.

The term probabilities indicate how specific, or common, different GO terms are and

Table 3.1: GO function evidence statistics. The percentage of annotations for different evidence types when using the gene products of the cell cycle pathway. The corresponding value for the entire genome is in brackets. Only gene products with known functional GO annotation are considered. There are 71 gene products and 111 annotations for the *S. cerevisiae* cell cycle, while there are 3997 gene products and 5983 annotations for the genome. The *H. Sapiens* cell cycle has 57 gene products and 152 annotations, while there are 20690 gene products and 39900 annotations for the genome. Abbreviations: TAS = traceable author statement, IDA = inferred from direct assay, IPI = inferred from physical interaction, ISS = inferred from sequence similarity, IEA = inferred from electronic annotation, IMP = inferred from mutant phenotype, ND = no biological data available, IGI = inferred from genetic interactions, IEP = inferred from expression pattern, NAS = non-traceable author statement, NR = not recorded, IC = inferred by curator.

Evidence	***S. cerevisiae***	***H. Sapiens***
TAS	53.2(26.3)	28.3(13.4)
IDA	24.3(24.4)	5.3(1.3)
IPI	8.1(7.9)	5.3(1.1)
ISS	6.3(24.8)	0(1.6)
IEA	0(0)	53.9(75.6)
IMP	5.4(12.2)	0(0.2)
ND	0(0)	0(<0.1)
IGI	1.8(3.4)	0(<0.1)
IEP	0.9(0.2)	0(0.1)
NAS	0(0.8)	3.9(5.3)
NR	0(0)	3.3(5.3)
IC	0(0)	0(<0.1)

the probabilities are used when the specificity of relation templates is derived. The specificity increases with decreasing term probability.

3.2.2 Binary relation extraction

The aim of binary relation extraction is to decompose the relations between protein complexes into simple binary relations between two gene products. Binary relations are subsequently used to derive templates. In the following procedure P-db is a regulatory pathway database (e.g. KEGG), C_i refers to a P-db complex containing a set of gene products and rel is a specific relation type, e.g. expression:

For each complex relation $C_i[rel]C_j$ in a P-db pathway:

Create $|C_i| \cdot |C_j|$ binary relations $GP_k[rel]GP_l$ by combining each gene product in C_i with each gene product in C_j, and add all new relations to a set MOD of model relations.

3.2.3 Template derivation

Templates representing generalised knowledge of regulatory relations are derived using the following procedure:

For each binary relation $GP_k[rel]GP_l \in MOD$:

Create templates $GOid_i[rel]GOid_j$ where $GOid_i \in S_k$=$\{GP_k$ terms with ascendants$\}$ and $GOid_j \in S_l$=$\{GP_l$ terms with ascendants$\}$,

yielding $|S_k| \cdot |S_l|$ templates. Add these templates to a set T.

For each template $GOid_i[rel]GOid_j \in T$:

GO-score $= 1 - \frac{p(GOid_i)+p(GOid_j)}{2}$

Sort T in descending order according to GO-score and remove duplicates.

For each template $t \in T$:

Record all binary pathway relations in MOD that conform to t.

The template GO-score is based on how common the GO terms are in the annotation database and it appears that a large number of templates get a GO-score close to 1. The histogram in figure 3.2 illustrates the distribution of GO-scores for templates

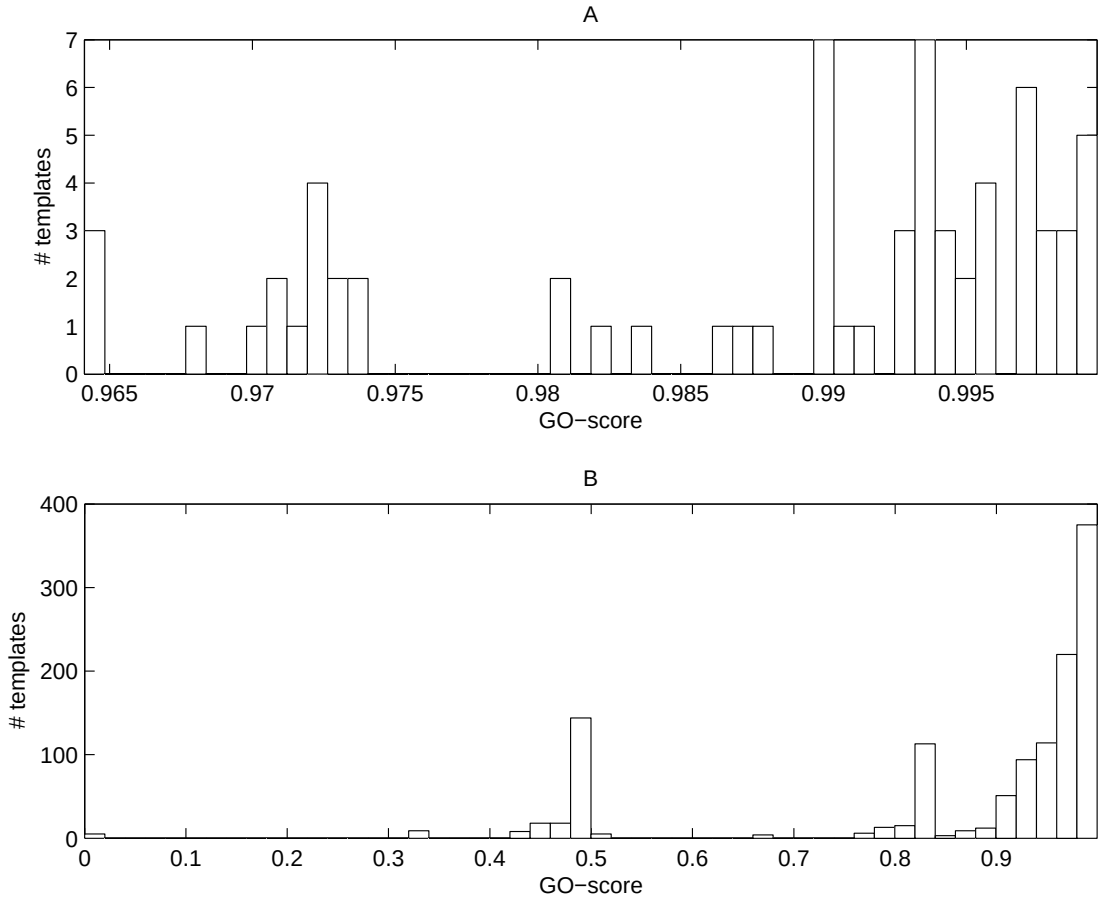


Figure 3.2: GO-score distribution for: (A) the 68 basic templates, i.e. templates at annotation level, and (B) the 1236 variant templates based on higher GO term abstractions.

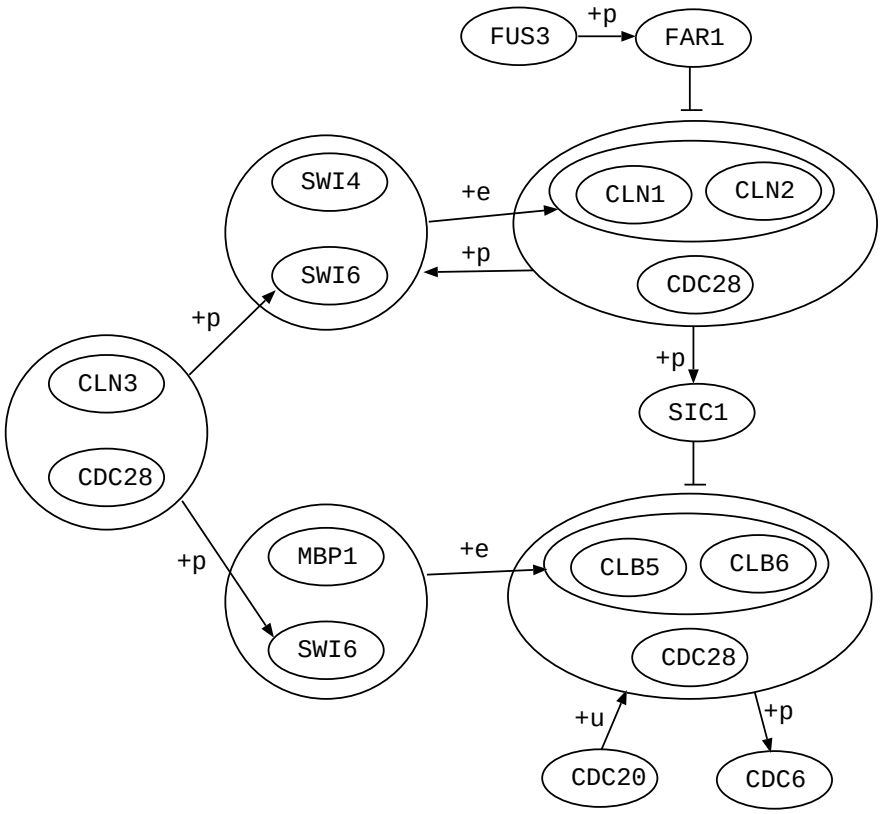


Figure 3.3: Part of the *S. cerevisiae* cell cycle pathway. +p = phosphorylation, +e = expression, +u = ubiquination, and T-shaped arrows represent inhibition.

derived from the *S. cerevisiae* cell cycle. A limited part of this pathway is shown in figure 3.3.

Only annotation level terms (terms directly associated with the gene products in A-db) are used to derive basic templates, and the score of the 68 templates vary in the interval $[0.96, 1[$. Templates containing GO terms of higher abstraction levels are referred to as variant templates, and the scores of these 1236 templates vary in the interval $[0, 1[$, but the majority of variant templates are in the same GO-score interval as the basic templates. As an example of template derivation, consider the relation "{SWI4,SWI6} [expression] {CLN1,CLN2}" in the *S. cerevisiae* cell cycle pathway in figure 3.3. This relation between gene product complexes is decomposed into the four binary relations "SWI4 [expression] CLN1", "SWI4 [expression] CLN2", "SWI6 [expression] CLN1" and "SWI6 [expression] CLN2". Each of these are used for template derivation. When "SWI4 [expression] CLN1" is used, each of the terms

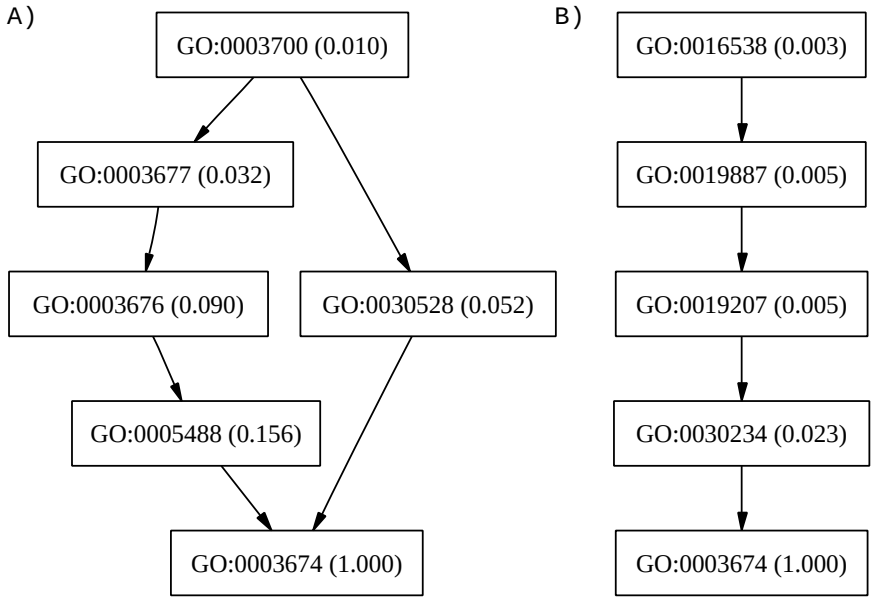


Figure 3.4: GO subgraphs for SWI4 (A) and CLN1 (B). GO terms are represented by nodes with their probability shown in brackets. Directed edges represent IS-A relationships. A) SWI4: GO:0003700 = transcription factor activity, GO:0003677 = DNA binding, GO:0003676 = nucleic acid binding, GO:0005488 = binding, GO:0003674 = molecular function, GO:0030528 = transcription regulator activity. B) CLN1: GO:0016538 = cyclin-dependent kinase regulator activity, GO:0019887 = protein kinase regulator activity, GO:0019207 = kinase regulator activity, GO:0030234 = enzyme regulator activity.

in the GO subgraph of SWI4 in figure 3.4 are combined with those for CLN1 to create templates. The basic template in this case would be "GO:0003700 [expression] GO:0016538", with a GO-score of 1-(0.010+0.003)/2 = 0.994. Variant templates are for example "GO:0003700 [expression] GO:0019887" and "GO:0030528 [expression] GO:0019207", with GO-scores 1-(0.010+0.005)/2 = 0.993 and 1-(0.052+0.005)/2 = 0.972, respectively. With six terms for SWI4 and five terms for CLN1, there is a total of $6 \cdot 5 = 30$ templates for this binary relation (one basic and 29 variant templates).

3.2.4 Hypothesis assessment

The set of templates derived in the previous subsection are subsequently used in the assessment of hypothetic regulatory relations derived using e.g. reverse engineering

algorithms. A hypothetic relation (or hypothesis for short) is defined as a directed relation between two gene products. There can be different relation types, e.g. expression and phosphorylation. The hypothesis assessment is described in the following procedure:

For each hypothesis $GPh_k[rel]GPh_l$ from a set H of regulatory hypotheses:
    For each template $GOid_i[rel]GOid_j \in T$:
        Report template match if $GOid_i \in \{GPh_k$ terms with ascendants$\}$
        and $GOid_j \in \{GPh_l$ terms with ascendants$\}$.

This procedure results in a list of templates conforming to each of the hypotheses, ranked in descending GO-score order. It is also possible to extend this procedure to report if the gene products in the hypothesis appear in any of the relations used to derive the template.

3.3 Results

This section presents a series of experiments which we use to assess the performance and show the utility of the proposed method for plausibility assessment.

3.3.1 On the complexity

This subsection elaborates on the time complexity of the GOTEM method. Referring to the method description in the previous section, the complexity of the GO term probability calculation is $O(|G|\cdot|GT_G|+|GT_{GO}|)$, where G is the set of gene products in the gene product annotation database, GT_G is the set of GO terms and their ascendant terms for a gene product in G, and GT_{GO} is the full set of gene product terms in the GO database. The binary relation extraction has $O(|R_c| \cdot |C|^2)$ complexity, where R_c is the set of relations between gene product complexes in the regulatory pathway database and C is the set of gene products in a complex in either the antecedent or consequent part of a relation in R_c. The template derivation part has a complexity of $O(|R_b| \cdot |S|^2 + |T| + |T| \cdot log|T| + |T| \cdot |MOD|)$, where R_b is the set of binary relations in the MOD set of model relations, S is the set of GO terms and its ascendants for a

gene product in the antecedent or consequent part of a binary relation in R_b, and T is the set of templates. The $|T| \cdot log|T|$ part is the complexity when sorting the templates using a quicksort algorithm. Finally, the hypothesis assessment has a complexity of $O(|H| \cdot |T| \cdot 2|GT_h|)$, where H is the set of hypothetical relations, and GT_h is the set of GO terms with ancestors in the antecedent or consequent part of a relation in H.

3.3.2 Pathways and their properties

Before explaining the specific experiments, we discuss pathways and their properties. KEGG pathways were used, and more specifically the KGML (XML) version of regulatory pathways. There are different types of relations between gene products in a regulatory pathway according to KEGG. In our study we chose to focus on activation, inhibition, expression, phosphorylation and dephosphorylation. The only relation used at the transcriptional level is expression, while the other four are at the post-translational level. Apart from these relation types KGML supports relations for repression (no such case present in the studied pathways), indirect effects, binding/association, dissociation, glycosylation, ubiquination and methylation.

Table 3.2 shows the distribution of binary relations and templates for different relation types, organisms and pathways. Two organisms (*S. cerevisiae* and *H. sapiens* and two pathways (cell cycle pathway and MAPK signaling pathway) are used in the experiments. The pathways were collected from KEGG on April 15^{th} 2004. Only binary relations where both gene products have known GO function are used. It can be noticed that there are no dephosphorylation relations in the MAPK signaling pathways and that the number of expression relations is smaller than for the cell cycle pathway. The main type of regulation is phosphorylation for the MAPK signaling pathway, which is quite different from the cell cycle pathway. The *H. sapiens* version of the MAPK signaling pathway contains a large number of binary relations and templates because relations exist between large gene product complexes in the pathway. The majority of the relations are at the post-translational level for the four pathways studied.

Table 3.2: Number of binary relations and templates for different relation types, organisms and pathways. Columns show combinations of organism and pathway (O1 = *S. cerevisiae*, O2 = *H. sapiens*, P1 = Cell cycle pathway, P2 = MAPK signaling pathway). Rows show relation type (Act = activation, Inh = inhibition, Exp = expression, Pho = phosphorylation, Dep = dephosphorylation). The table contains [number of binary relations] / [number of templates] ([number of basic templates]), except for the last row which shows the total number of gene products involved in the different pathways.

	O1P1	O1P2	O2P1	O2P2
Act	8/122 (9)	12/217 (19)	7/713 (62)	323/2236 (301)
Inh	29/419 (22)	8/193 (5)	47/1278 (112)	211/1378 (183)
Exp	11/102 (7)	5/87 (7)	10/396 (21)	3/70 (8)
Pho	37/526 (25)	25/599 (21)	47/884 (93)	321/2767 (462)
Dep	4/135 (5)	0/0 (0)	10/180 (14)	0/0 (0)
#GPs	71	38	57	227

3.3.3 *S. cerevisiae* cell cycle pathway

A diagnostic experiment was designed, which aims to investigate if the GO-score of templates can be used to discriminate between a set MOD of biologically plausible relations, and a disjoint hypothesis set EXT sharing the gene products of MOD, representing potentially implausible relations. MOD is used as a "golden standard" of regulation.

The receiver operating characteristic (ROC) curve can be used to assess the trade-off between sensitivity and specificity of a diagnostic algorithm (Swets et al. 2000), and has e.g. been used by Husmeier (2003) to investigate the effects of different parameter settings when inferring regulatory networks. The true positive rate TPR is plotted against the false positive rate FPR for a number of GO-score thresholds that are required for a hypothesis match. TPR is defined as (Swets et al. 2000):

$$TPR = \frac{TP}{TP + FN} \tag{3.1}$$

where TP is number of true positives and FN is number of false negatives. The false positive rate is calculated as

$$FPR = \frac{FP}{FP + TN} \tag{3.2}$$

where FP is number of false positives and TN is number of true negatives.

The gene products in MOD are used to create an extended relation set EXT containing all possible $N(N-1)$ non-reflexive relations between the N gene products in MOD minus the relations actually present in MOD. Hence, MOD and EXT are disjoint sets of relations. MOD is created from the *S. cerevisiae* cell cycle, where there are 71 gene products of known function and 89 binary relations. This results in 4881 binary relations for EXT. Each binary relation in the MOD and EXT sets is matched against the template set and the GO-score of the top template is recorded. The GO-scores for the top templates matching the hypotheses are used to calculate TP, FN, FP and TN. Figure 3.5 shows the number of relations as a function of a required GO-score threshold for the relation-associated top scoring templates of the MOD (top) and EXT (bottom) relation sets. It can be observed that all the 89 MOD relations are maintained until the GO-score threshold reaches approximately

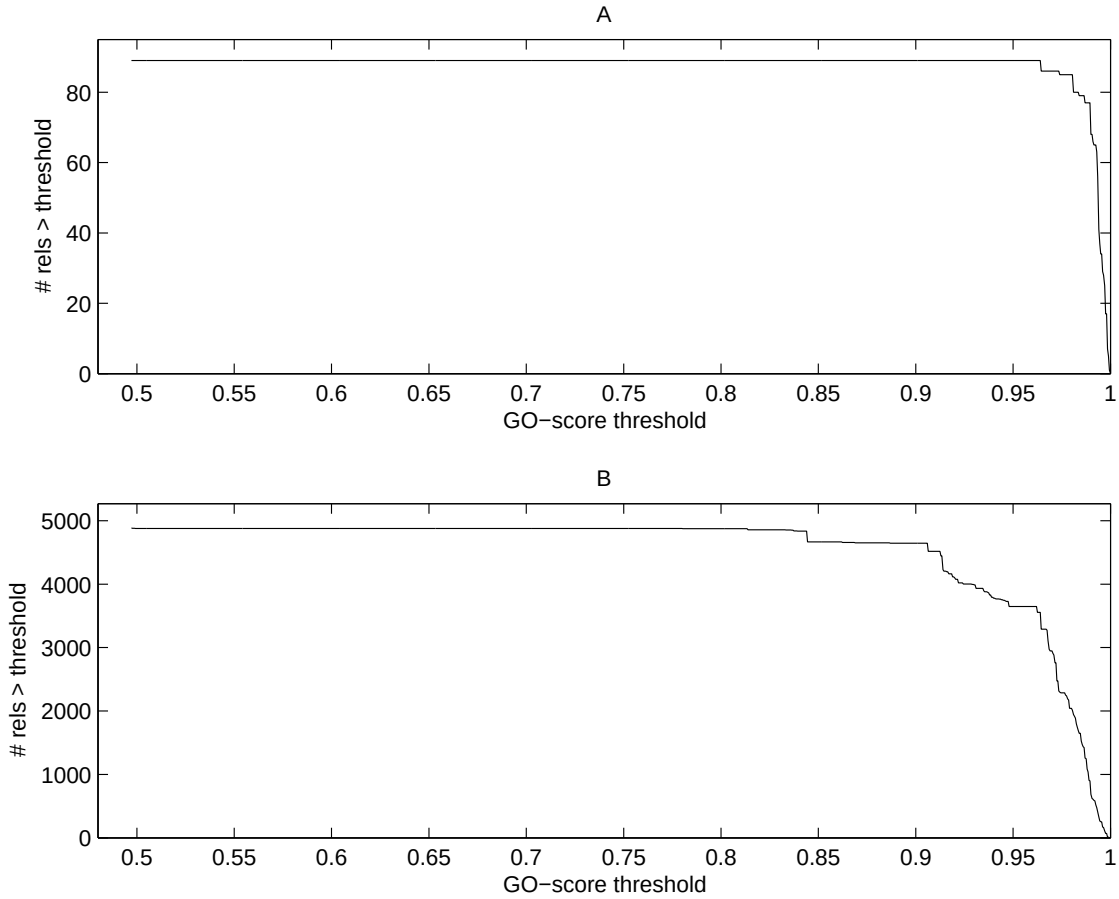


Figure 3.5: Number of relations as a function of GO-score threshold for: (A) the MOD relation set, and (B) the EXT set.

0.96. In contrast, the point where the first EXT relation is lost comes as early as at GO-score $\approx$ 0.5, and starting at 0.8 the number of EXT relations are reduced considerably.

The curves in figure 3.5 are sampled linearly with 2000 points from the point before the first relation is lost to the end of the GO-score threshold axis. In the case of the *S. cerevisiae* cell cycle the interval is [0.50,1], resulting in the vectors $\overrightarrow{M}$ for MOD and $\overrightarrow{E}$ for EXT. We define the number of true positives (TP_i) for a specific GO-score threshold i, false negatives (FN_i), false positives (FP_i) and true negatives (TN_i) as

$$TP_i \quad = \quad M_i \tag{3.3}$$

$$FN_i = max(\overrightarrow{M}) - M_i \tag{3.4}$$

$$FP_i = E_i \tag{3.5}$$

$$TN_i = max(\overrightarrow{E}) - E_i \tag{3.6}$$

The vectors $\overrightarrow{TPR}$ and $\overrightarrow{FPR}$ are derived using equations 3.1 and 3.2. Plotting $\overrightarrow{TPR}$ against $\overrightarrow{FPR}$ results in an ROC curve. The diagnostic performance can be summarised by calculating the area under the ROC curve (Swets et al. 2000), which requires some processing. Duplicate pairs $\{TPR_i, FPR_i\}$ are removed, and in cases where there are several TPR values for one specific FPR value, the average TPR value is used. Finally, a 500-point linear interpolation is performed on the ROC curve. The area of the curve is calculated as

$$A = \sum_{i=1}^{N} TPR_i \cdot \Delta_{FPR} \tag{3.7}$$

where N is total number of TPR values and Δ_{FPR} is the absolute difference in FPR between two consecutive points. The described procedure generates the ROC curve in figure 3.6 which has an area of 0.886. As the ROC area $A \in [0, 1]$, this result shows that a large proportion of the potentially implausible EXT relations are filtered out while a large proportion of the plausible MOD relations are maintained.

3.3.4 Several organisms and pathways

The results in figures 3.5 and 3.6 use the same set of binary relations for template derivation and model, which raises the question whether a set of templates can be used to assess models related to other pathways and organisms. For this reason a test matrix according to table 3.3 is set up. Apart from the *S. cerevisiae* cell cycle pathway, the *H. sapiens* cell cycle is used and also the MAPK signaling pathway for both organisms. The four pathways result in 16 different combinations if one pathway is used for template derivation and one for the model. The ROC areas along the diagonal in table 3.3 are large (in the interval $[0.846, 0.890]$), i.e. the method does well when the same pathway and organism is used for both template derivation and model. If the same organism but different pathways are used for templates and

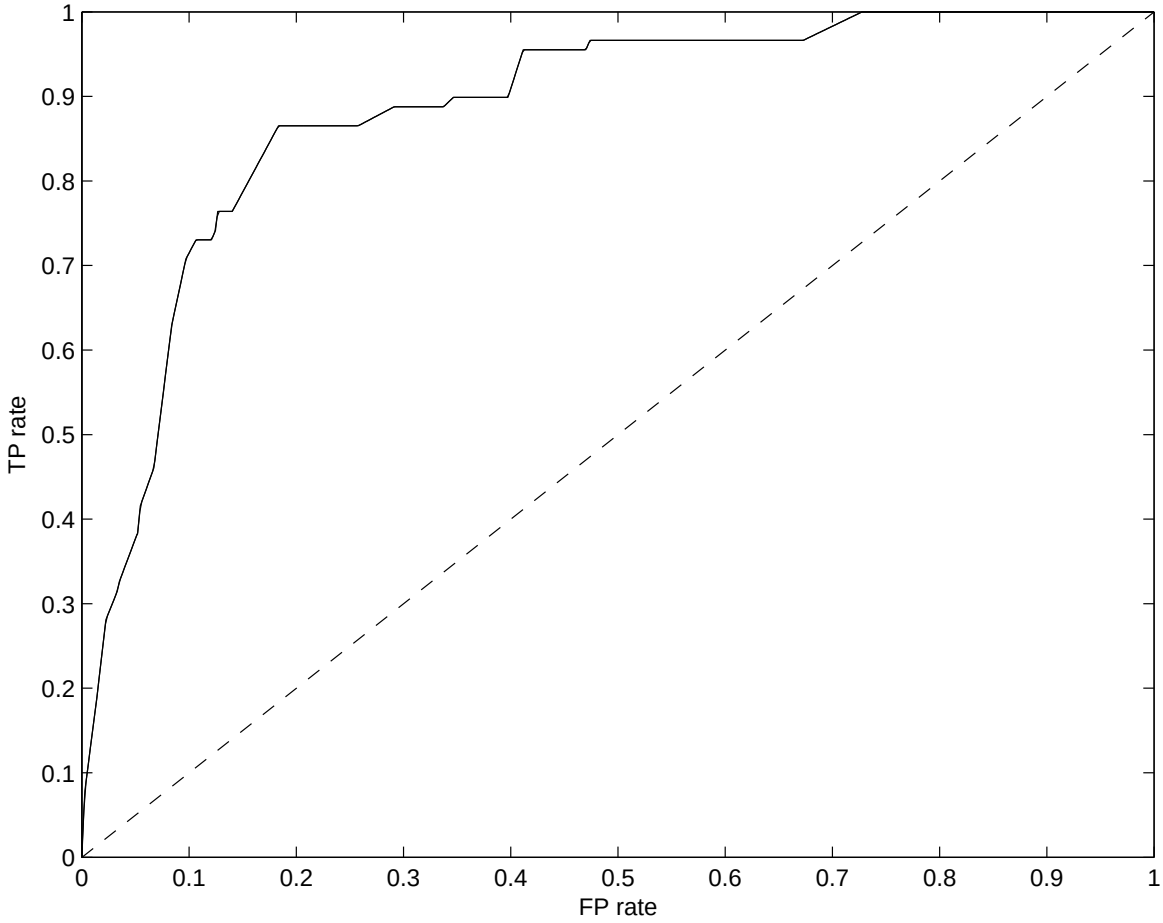


Figure 3.6: ROC curve based on the relation curves in figure 3.5. Area = 0.886

Table 3.3: ROC area results. Rows show organism and pathway used for template derivation, and columns show organism and pathway used as model. O1=*S. cerevisiae*, O2=*H. sapiens*, P1=Cell cycle pathway, P2=MAPK signaling pathway

	O1P1	O1P2	O2P1	O2P2
O1P1	0.886	0.648	0.682	0.690
O1P2	0.562	0.846	0.634	0.710
O2P1	0.615	0.622	0.860	0.677
O2P2	0.473	0.701	0.615	0.890

model, the ROC area is generally smaller. As an example, the cell cycle pathway of *S. cerevisiae* for template derivation and the MAPK signaling pathway of *S. cerevisiae* as model, produced an ROC area of 0.648. An even smaller area results if the pathways used for templates and model are switched, e.g. 0.562 for *S. cerevisiae*. This indicates that the cell cycle pathway contains certain types of relations that are not available in the MAPK signaling pathway, which results in that templates derived from the MAPK signaling pathway can not capture certain cell cycle relations. An ROC area of around 0.5 means that the algorithm is incapable of discriminating between the MOD relations and the EXT relations, i.e. no better than random guessing (Swets et al. 2000). However, the small ROC areas obtained when using different types of pathways for template and model can be seen as an advantage, since hypothetic relations are less related to the specific pathways used to derive the templates.

Table 3.3 also shows results from using a pathway from one organism to derive templates and a pathway from another organism for the model. The area varies in the interval [0.615,0.710] for 7 cases out of 8, which represents approximately the same performance as when a different pathway is used for templates and model for the same organism. The exception is an area of 0.473, obtained when the MAPK signaling pathway for *H. sapiens* is used to derive templates and the cell cycle pathway for *S. cerevisiae* is used as model. The last case represents performance approximately equivalent to random guessing. The results also show that the area is smaller when the same pathway for two different organisms is used for templates and model, when comparing with the case where the same pathway for one organism is used.

3.3.5 Effects of subdividing a pathway

In addition to the experiments according to table 3.3 the set of binary relations MOD of the *S. cerevisiae* cell cycle was subdivided randomly into two equally sized disjoint sets: MOD_T used for template inference and MOD_M used for the sets MOD and EXT. Each type of relation was treated individually during the subdivision, i.e. half of the relations are extracted from e.g. "expression" and "phosphorylation" individually. The reason for this is obvious: it is not possible to detect relations of a certain type if no relation of this type is used for template derivation. Furthermore,

the MOD and EXT sets are created in the same way as described before except that MOD_M is used instead of MOD. 10 random subdivisions using a uniform distribution are performed, and the mean ROC area is 0.761 with a maximum area of 0.824 and a minimum area of 0.686. The main reason for the relatively large deviations is probably the small number of relations in the pathway and that the different terms in those relations can be unevenly distributed between MOD_T and MOD_M.

Another experiment is performed for *S. cerevisiae* where both pathways are used for templates and both pathways for the model. This results in an area of 0.877, i.e. almost identical to the areas obtained when only one pathway is used for both templates and model. Performing 10 random subdivisions into equally sized subsets MOD_T and MOD_M of the set of relations from both pathways results in a mean ROC area of 0.764 with a maximum of 0.793 and a minimum of 0.747, which is approximately the same as for the subdivision experiment when only the cell cycle pathway is used. One reason for the smaller variation in area is that the number of relations used for templates is larger, resulting in a set of templates MOD_T that is more likely to successfully match the relations in MOD_M.

3.3.6 Assessing reverse engineered hypotheses

We also study sets of hypothetical relations derived by applying a dynamic Bayesian network technique to gene expression data measuring the expression of genes during the *S. cerevisiae* cell cycle (Bergmann-Sigurdsteinsdottir 2004, Kim et al. 2003). The hypotheses of Bergmann-Sigurdsteinsdottir (2004) are presented together with their associated top scoring templates in table 3.4, sorted in descending order according to GO-score. In order to facilitate interpretation of GO-score values and make it easier to set a threshold for which hypotheses to consider plausible we also calculate an "expect" value E for each hypothesis H_i according to:

$$E(H_i) = |H| \cdot \frac{|\{t : t \in T \wedge GS(t) \geq GS(H_i)\}|}{|T|} \tag{3.8}$$

where H is the hypothesis set, T is the template set and GS is GO-score. $E(H_i)$ is the expected number of template hits for H_i with $GS(t) \geq GS(H_i)$, given that templates are randomly drawn from T. In addition, each hypothesis in table 3.4 is classified

into one of four different categories of conformance with respect to the target network in figure 3.3 (Kim et al. 2003); correct, transitive, misdirected and incorrect. The hypothetical relations are limited to cover only the part of the cell cycle shown in figure 3.3. The relations in the whole *S. cerevisiae* cell cycle pathway are used to derive the templates. The reverse engineering algorithm finds the hypothetic relations in table 3.4 plausible with respect to the gene expression data. The GO-score represents the level of biological plausibility with respect to the set of regulatory relations used for template derivation.

The GO-score results in table 3.4 are visualised in figure 3.7 where the accumulation of hypotheses from different hypothesis classes as a function of descending logarithmic GO-score is presented. Logarithmic GO-score is used for presentation purposes and is calculated as $-log_2((p(GOid_i) + p(GOid_j))/2)$.

It can be noted that the curve representing the "correct" class of hypotheses reaches 100% before any other class, and the "incorrect" class reaches this level as the last class. This suggests that templates and their GO-scores are useful in the evaluation of hypothesis sets derived from microarray gene expression data mining. Kim et al. (2003) also used dynamic Bayesian networks to derive 16 regulatory hypotheses (3 correct, 5 transitive, 4 misdirected and 4 incorrect). Figure 3.8 corresponds to figure 3.7 but for the new set of hypothetic relations. The results are similar; the "correct" class accumulates all hypothetical relations before any other class and the "incorrect" class finishes last.

3.3.7 On the similarity of gene products

In table 3.4, some subsets of hypotheses share the same template and GO-score, e.g. {H3,H5} and {H6,H7}. H3 and H6 are correct, H5 is misdirected and H7 is incorrect with respect to the current knowledge of the pathway in figure 3.3. The reason for this is that the gene products with respect to GO functional annotation are identical. The templates matching these four relations are basic, i.e. derived from annotation level terms and are therefore in a sense more specific than variant templates, even if the GO-score is the same or lower. An example variant template in table 3.4 is the one for the relation set {H13,H14,H15,H16} with a GO-score of 0.9960 which is higher

Table 3.4: Regulatory hypotheses from Bergmann-Sigurdsteinsdottir (2004) and best matching templates. The *Hypothesis* column shows a directed hypothetical relation between two gene products. The first number in the *Template* column shows the informative digits in the GO term identifier for the left hand side of the template, coding for a specific molecular function. In the same column within square brackets is the type of relation for the template: E = expression, P = phosphorylation, I = inhibition. The next number is the GO term identifier for the right hand side of the template, followed by template type within brackets: B = basic template, V = variant template. The *Score* column shows the GO-score of the template. The value in the *E* column is the expected number of hits with a GO-score value greater than or equal to the matching template GO-score. The hypothesis class is shown in the last column: C = correct edge with respect to the known model. Relations containing gene products that are situated in the same complex are regarded as correct. T = transitive edge, skipping one gene product. M = misdirected edge. I = incorrect edge. Hypothesis classes "Cl" according to Kim et al. (2003). Note that all five correct hypotheses had the lowest average *E*, followed by transitive hypotheses, while mis-directed and incorrect hypotheses had substantially higher average E (C: 0.62, T: 0.96, I: 3.07, M: 3.48)

	Hypothesis	**Template**	**Score**	**E**	**Cl**
H1	FUS3→SIC1	4707[P]19210(V)	0.9994	0.0736	I
H2	CDC28→CDC6	4693[P]3689(B)	0.9990	0.1840	C
H3	CLN1→SIC1	16538[P]19210(B)	0.9981	0.4669	C
H4	SIC1→CLN2	19210[I]16538(V)	0.9981	0.4669	M
H5	CLN2→FAR1	16538[P]19210(B)	0.9981	0.4669	M
H6	CLB5→CDC6	16538[P]3689(B)	0.9980	0.5153	C
H7	CLN3→CDC6	16538[P]3689(B)	0.9980	0.5153	I
H8	FAR1→FAR1	4861[I]19887(V)	0.9974	0.5890	I
H9	FAR1→SIC1	19887[P]19210(V)	0.9973	0.6258	T
H10	FUS3→CLN3	4707[P]19887(V)	0.9973	0.6258	I
H11	CLB5→CDC28	19887[I]4693(V)	0.9969	0.9571	C
H12	CDC28→CLB5	4693[P]19207(V)	0.9967	0.9939	C
H13	CLN1→CLB6	19887[I]16538(V)	0.9960	1.2883	T
H14	CLB6→CLN2	19887[I]16538(V)	0.9960	1.2883	I
H15	CLN1→CLN3	19887[I]16538(V)	0.9960	1.2883	I
H16	CLB6→CLN1	19887[I]16538(V)	0.9960	1.2883	I
H17	CLN2→SWI4	16538[P]3700(B)	0.9934	2.3558	M
H18	FUS3→SWI4	4674[P]3700(V)	0.9895	3.5890	I
H19	CDC28→FUS3	4674[E]4674(V)	0.9890	3.6810	I
H20	CDC20→FAR1	30234[P]19210(V)	0.9880	4.0307	I
H21	CDC20→CLN2	30234[I]16538(V)	0.9867	4.6012	I
H22	CDC6→CLB6	3677[E]16538(B)	0.9823	7.0675	M
H23	CDC6→CLB5	3677[E]16538(B)	0.9823	7.0675	M
H24	SWI6→SWI6	30528[E]5515(V)	0.9479	12.865	I

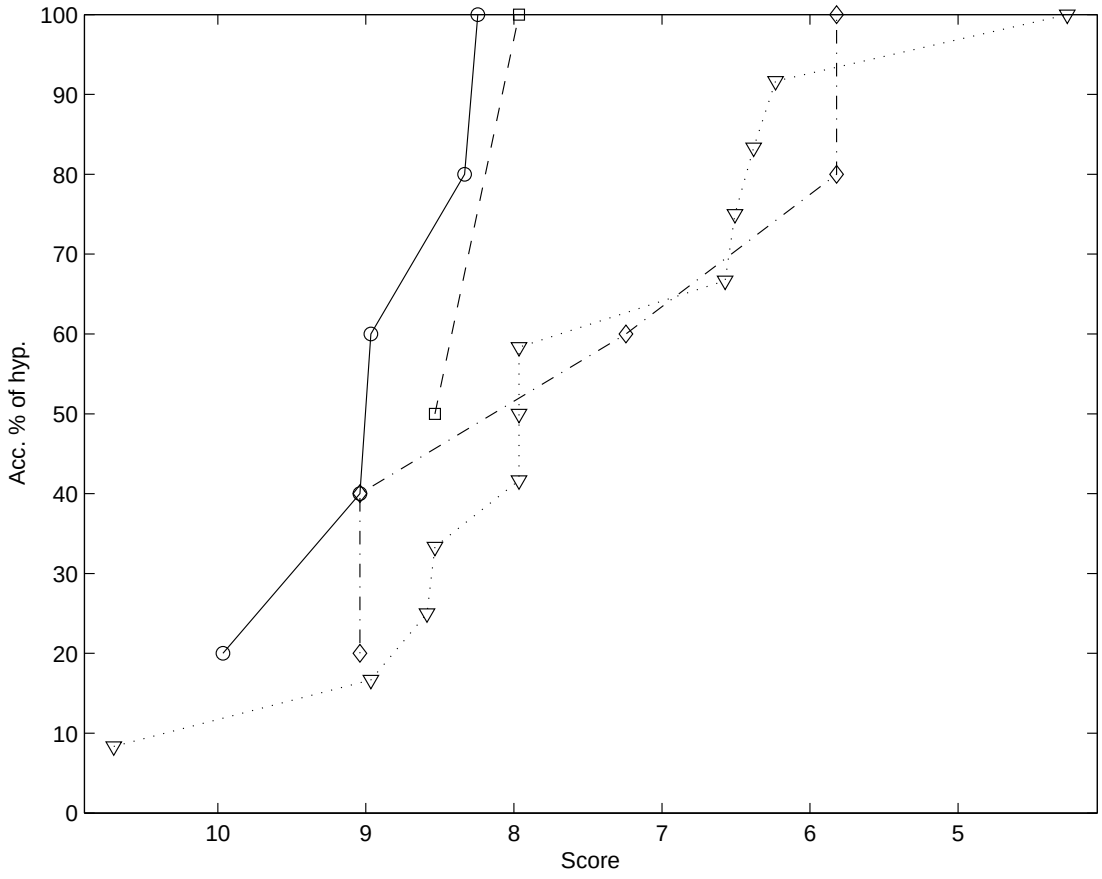


Figure 3.7: The percentage of accumulated hypotheses as a function of logarithmic GO-score for different hypothesis classes for the relations derived by a dynamic Bayesian network technique in table 3.4 (Bergmann-Sigurdsteinsdottir 2004). Figure legend: circle = correct, square = transitive, diamond = misdirected, triangle = incorrect.

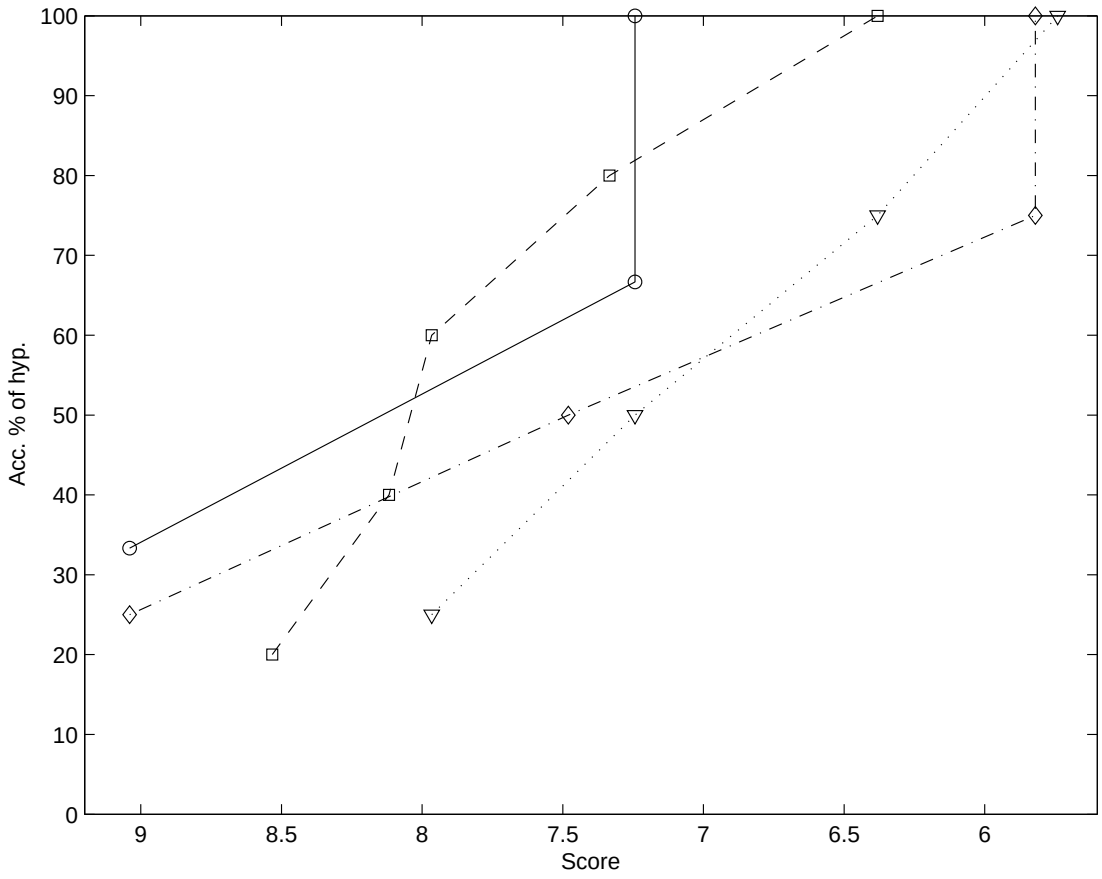


Figure 3.8: The percentage of accumulated hypotheses as a function of logarithmic GO-score for different hypothesis classes for relations derived by a dynamic Bayesian network technique (Kim et al. 2003). Figure legend is the same as for figure 3.7.

than the GO-score of 0.9823 for the basic template matching {H22,H23}.

It can also be discussed whether it is desirable that some hypotheses belonging to the class "incorrect" are assigned GO-scores as high or even higher than some hypotheses of the "correct" class. But as Husmeier (2003) points out, a hypothesis is not necessarily incorrect because it is not in the model. Discarding a hypothesis would require additional multiple gene knockout experiments. This makes the concept of biological plausibility attractive, and that templates and their GO-score indicate the degree of plausibility with respect to the knowledge used to derive the templates.

To illustrate the semantic similarity between the 14 gene products in figure 3.3, we perform an analysis using the Phylip software (Felsenstein 1993) and the semantic distance measure proposed by Lin (1998) and applied to Gene Ontology terms by Lord et al. (2003b). The measure of semantic similarity between ontology terms is defined as

$$sim(t_1, t_2) = \frac{2 \cdot \ln\, p_{ms}(t_1, t_2)}{\ln\, p(t_1) + \ln\, p(t_2)} \tag{3.9}$$

where $p(t_x)$ is the probability of term t_x and $p_{ms}(t_1, t_2)$ is the probability of the minimum subsumer for terms t_1 and t_2. The minimum subsumer is the ancestor term with lowest probability that is common to both t_1 and t_2. As we are interested in calculating the similarity between gene products rather than ontology terms and a gene product can be annotated with several terms, the average term to term similarity is used (Lord et al. 2003b). Phylip requires a matrix of distances between the gene products and the Lin similarity measure results in values in the interval [0,1] where 1 represents identical terms. For this reason the semantic distance between terms is defined as

$$dist(t_1, t_2) = 1 - sim(t_1, t_2) \tag{3.10}$$

The neighbour joining algorithm is applied to the 14 gene products and the resulting tree is shown in figure 3.9. FUS3 and CDC28 are both involved in protein kinase activity and appear close to each other. MBP1 and SWI4 are both annotated with transcription factor activity, whereas SWI6 has the annotations transcription co-activator activity and protein binding which explains the distance to MBP1 and SWI4. CDC6 has annotations protein binding and DNA clamploader activity, hence being somewhat similar to SWI6. SIC1 has the annotation kinase inhibitor activity

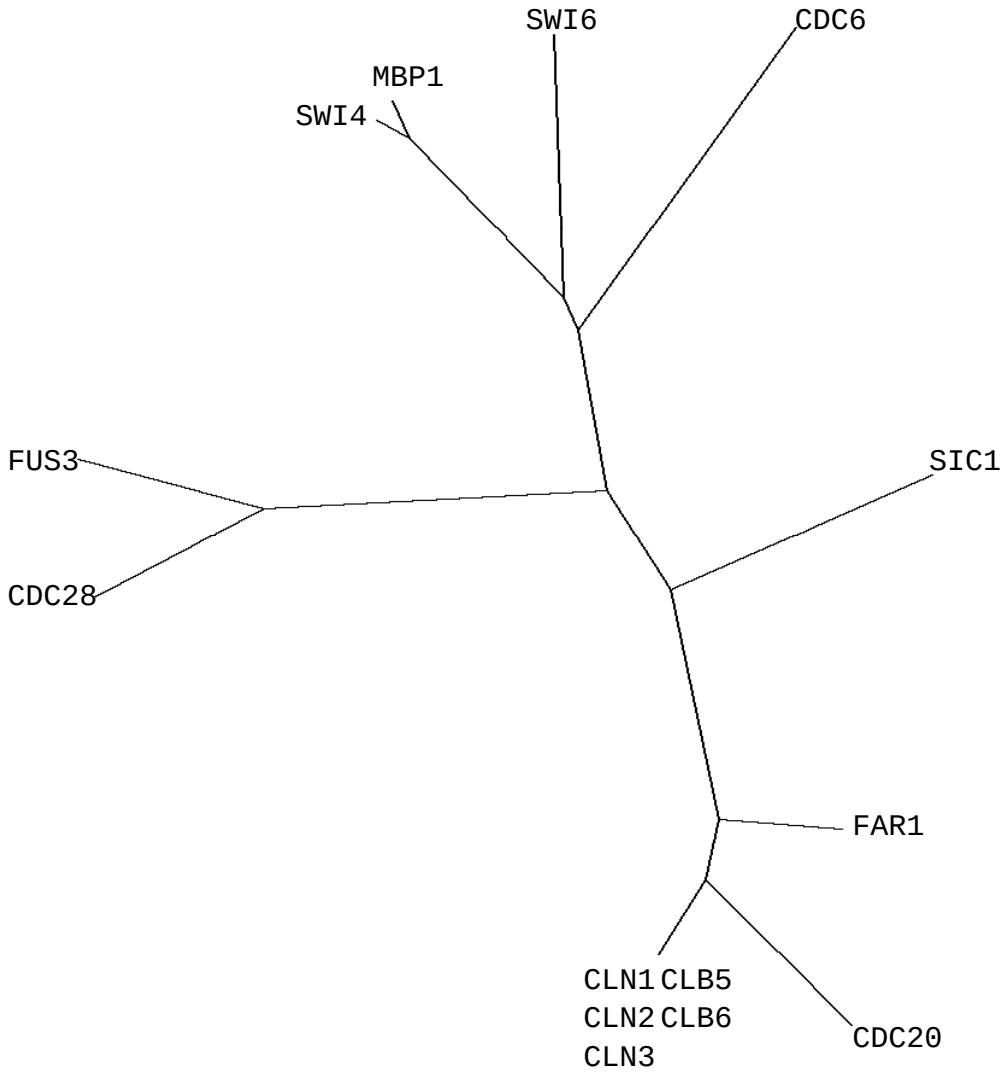


Figure 3.9: Tree derived using neighbour joining based on functional semantic distances for the 14 gene products shown in figure 3.3.

and is not so similar to any other of the gene products. Five gene products (CLN1, CLN2, CLN3, CLB5 and CLB6) have exactly the same annotation; cyclin-dependent protein kinase activity and therefore appear at the same position in the tree. CDC20 is annotated with enzyme activator activity and the annotation of FAR1 is cyclin-dependent protein kinase inhibitor activity.

Hence, many relations can be matched to one template even if it is at the basic level, and this is due to the limited granularity of GO terms and the fact that many gene products share the same GO annotation.

3.4 Discussion

In this chapter we propose a systematic method based on general knowledge of regulatory pathways for assessing the biological plausibility of hypotheses derived during regulatory network reconstruction. Our results demonstrate that the method is able to filter out a large proportion of potentially implausible hypotheses, thus greatly improving the specificity of the regulatory network reconstruction process.

Gene products in some hypotheses are identical or very similar to gene products in relations used for template derivation, with respect to GO functional annotation. This means that the hypotheses are highly feasible, according to our current general knowledge, while at the same time they appear to be incorrect, according to our current specific knowledge. It is possible that a future more fine-grained GO annotation allows us to better distinguish these hypotheses. It is also possible that future experiments show that some of these hypothetic relations actually exist.

The results show that our method performs best when similar relations and gene products are used for template and hypothesis derivation. However, we expect that the generality of the method will improve if a more diverse set of pathways is used for template derivation. Furthermore, the number of regulatory pathways in KEGG is small, but additional pathway sources such as ResNet (http://www.ariadnegenomics.com) and BioCarta (http://www.biocarta.com) can extend the knowledge base.

Our approach decomposes relations between gene product complexes into atomic binary relations between individual gene products. This leads to certain problems

as can be observed e.g. for the CLN3-CDC28 complex in figure 3.3; CLN3 regulates the CDC28 kinase but the complex as a unit phosphorylates SWI6. Hence, it is not entirely true that CLN3 and CDC28 individually phosphorylate SWI6 as our method suggests. On the other hand it is partially true because CLN3 helps to regulate SWI6 but not on its own. A possibility is to extend our method so that complex relations can be handled without being decomposed into binary relations.

Our method aims to automatically identify the most biologically plausible hypotheses by a measure based on GO term specificity, but it would also be beneficial to have the top scoring hypotheses assessed by a domain expert in order to reduce the number of false positives. As was shown in the results section, it is often the case that basic templates derived from GO terms at the annotation level have lower scores than variant templates at higher abstraction levels. Different ways of scoring templates could be studied. A way to get better accuracy for our method could also be to incorporate other sources of biological knowledge such as databases containing transcription factor binding site information and protein interactions. In this chapter we used GO annotations of all evidence types. It would therefore be of interest to examine the impact on performance when different evidence types are used.

For the sake of method applicability, a crucial question is what kind of regulatory relations that are likely to be observable from gene expression data. It has been claimed that it is only regulation at the transcriptional level we are likely to observe, i.e. regulatory proteins that control the transcription rate of mRNA for genes of an organism (Husmeier 2003, Knudsen 2002). This would correspond only to the expression relation type in our model. But protein to protein relations could be derived by also using proteomics related methods and knowledge such as gene interaction maps, rather than only gene chip technology (Knudsen 2002).

Chapter 4

GOSAP: Gene Ontology based Semantic Alignment of Biological Pathways

We present a new method for semantic comparison of biological pathways, aiming to discover evolutionary conservation of pathways between species. Our method uses all three sub-ontologies of Gene Ontology (GO) and a measure of semantic similarity to calculate match scores between gene products. These scores are used for finding local pairwise pathway alignments. This approach has the advantage of being applicable to all types of pathways where nodes are gene products, e.g. regulatory pathways, signalling pathways and metabolic enzyme-to-enzyme pathways. We demonstrate the usefulness of the method using regulatory and metabolic pathways from *E. coli* and *S. cerevisiae* as examples.

4.1 Introduction

A large number of biological pathways are being elucidated for many different organisms such as *S. cerevisiae* and *E. coli*, and these are stored in various databases such as KEGG (Kanehisa and Goto 2000), EcoCyc (Karp et al. 2004) and MetaCyc (Caspi et al. 2008). Due to the availability of such databases, there is a need for algorithms capable of searching for homologues to pathway queries in a collection of known pathways (Pinter et al. 2005). These algorithms should also return alignments between matching pathway fragments. Furthermore, these pathway alignment methods should rely on approximate, rather than exact, matching in biological pathways (Pinter et al. 2005, Koyutürk et al. 2004). One reason is that approximate matching can associate gene products that have different labels but are known to perform similar tasks in the cell.

The previous work on this topic can be divided into three categories: 1) approaches performing pathway comparison or assessment without making alignments, 2) methods that align pathways without performing any semantic generalisation, instead just matching gene product labels or using approximate matching by sequence similarity, 3) methods for pathway alignment that perform semantic generalisation using abstraction hierarchies or ontologies. As will be shown, our work belongs to the third category.

As an example approach from the first category, Ogata et al. (2000) proposed a method for detection of functionally related enzyme clusters, where topological properties of metabolic pathways are considered. As another example, an approach for detecting frequent modules and patterns in biological pathways has been reported by Koyutürk et al. (2004). However, their method is geared towards increasing the computational tractability by simplifying large biological pathways into smaller subpathways and does not directly address alignments. Berg and Lässig (2004) described a method for topological motif search in biological pathways, which they applied to the gene regulation network of *E. coli*.

A number of methods in the second category have also been proposed, i.e. methods that align pathways by exact matching of gene product labels, without performing

any semantic generalisation using abstraction hierarchies or ontologies. Dandekar et al. (1999) presented an approach for pathway alignment which involved analysis and comparison of biochemical data, metabolic pathway analysis using the elementary modes concept, and comparative analysis of a set of completely sequenced genomes where EC terms were used to categorise results. Furthermore, a method using both topology and sequence similarity to derive alignments between protein interaction pathways has been proposed (Kelley et al. 2003), where conserved interaction pathways were identified for *S. cerevisiae* and *H. pylori*. An extended version of this method has also been developed for multiple pathway alignment, i.e. involving more than two pathways at a time (Sharan et al. 2005). Koyutürk et al. (2005) proposed a framework where pairwise local alignment is used to understand the evolution of protein interaction networks. This approach uses theoretical models of protein interaction network evolution in order to derive conserved interaction patterns. Similarly, The Graemlin algorithm for alignment of multiple protein interaction networks also detects conserved functional modules (Flannick et al. 2006), but offers a computationally efficient algorithm which scales well for an arbitrary number of networks. Berg and Lässig (2006) proposed a Bayesian alignment algorithm, which considers commonalities between interaction networks of different species using measures based on both topology and sequence similarity. QPath (Shlomi et al. 2006) is another algorithm for searching a protein interaction network for homologies. A simple query path is aligned by dynamic programming to paths in the PIN using a scoring scheme involving the sequence similarity, PIN interaction reliability, and the difference in protein insertions and deletions. Additionally, the functional enrichment of pathway hits was calculated using GO process annotation of the proteins. An exact polynomial time algorithm for matching a biological query subgraph to a subgraph in a model network was proposed recently (Yang and Sze 2007), with a scoring scheme involving the logarithm of the BLAST E-value (i.e. sequence similarity). It is shown that the algorithm is applicable to both undirected interaction networks and directed metabolic networks. SAGA (Tian et al. 2007) is another subgraph matching tool where the scoring is based on structural distance between query and model graph, node mismatch, and penalties for node gaps. However, this tool does not consider that two proteins can be similar

even if they do not have the same label.

In the third category, we find a small number of approaches that are the most similar to the one proposed in this chapter, since they also derive semantic pathway alignments using abstraction hierarchies or ontologies. A method for deriving multiple alignments of paths in metabolic pathways has been proposed by Tohsato et al. (2000), where the hierarchy of the enzyme nomenclature is used for generalising about enzymes. In addition, a method for alignment of metabolic pathways using a technique known as approximate labeled sub-tree homeomorphism, has been proposed by Pinter et al. (2005), where the hierarchy of the enzyme nomenclature once again is used for generalisation. This approach combines the topological and semantic properties of metabolic enzyme-to-enzyme pathways. The method was tested by performing intra- and interpathway comparisons for all metabolic pathways available in the SGD (Saccharomyces Genome Database) for *S. cerevisiae* and *E. coli*. An approach that improved on the work of Pinter et al. (2005) was proposed by Wernicke and Rasche (2007), which is more efficient and is not restricted to tree topologies.

Here, we propose GOSAP, a local alignment method that uses Gene Ontology (GO) (Ashburner et al. 2000) to compare paths in biological pathways. To our knowledge, GO has not previously been used for deriving semantic alignments of paths in biological pathways. The main advantage of using GO to calculate similarity scores is that it enables semantic analysis of pathways where nodes are not only enzymes, but any kind of gene product. Hence, our proposed method is applicable to all types of biological pathways where nodes are gene products, e.g. regulatory pathways, signalling pathways and metabolic enzyme-to-enzyme pathways. Another novelty is the use of combined alignment scores involving all three sub-ontologies of GO. The EC hierarchy, which has been used in related work, classifies enzymes into categories of catalysed reactions, which is a functional classification. GO, on the other hand, covers function as well as the biological process and cellular localisation of gene products. Hence, a wider spectrum of biological properties is reflected in the match scores, which is expected to improve the sensitivity and biological relevance of the resulting alignments. GOSAP is published in Gamalielsson and Olsson (2008b), and was prior to this filed as a technical report (Gamalielsson and Olsson 2005b).

4.1.1 Related work

An approach to semantic multiple alignment of enzyme paths is described in Tohsato et al. (2000). This was the first paper where the enzyme nomenclature classifications were used for performing semantic comparisons in a pathway scenario. In an adjacent research area, pathway clustering, it was earlier common to compare enzymes by their sequence similarity. However, as Galperin et al. (1998) point out, it is common that two enzymes seem unrelated in terms of protein sequence even when they share the same protein structure. This motivated the choice of the enzyme nomenclature classification. The enzyme hierarchy has 4 levels, and at the first level there are six broad classes of enzymatic activity; oxidoreductase, transferase, hydrolase, lyase, isomerase and ligase. The second level specifies what the top enzyme class acts on (e.g. a peptide bond). The kind of acceptor (e.g. NAD+) is described at the third level. The lowest level specifies a certain reaction. The similarity $I(h)$ between two enzymes E_1 and E_2 sharing a common upper enzyme classification h is calculated as $I(h) = log_2(1/C(h)) - log_2(p(h))$, where $C(h)$ is the number of enzymes at or under level h. $p(h)$ denotes the occurence probability of h and is defined as $p(h) = \sum_{i=1}^{n} o(h, s_i) / \sum_{i=1}^{n} N(s_i)$, where s_i belongs to a specified set of pathways $S = \{s_1, ..., s_n\}$, $o(h, s_i)$ is the number of occurences of h in s_i, and $N(s_i)$ is the total number of classifications in s_i. A modified Needleman-Wunsch global alignment algorithm was described for pairwise path comparisons. Instead of using a substitution matrix, the semantic score $I(h)$ for comparing two enzymes was used together with a linear gap penalty of -15. Additionally, a greedy multiple alignment algorithm similar to the Feng and Dolittle (1987) algorithm, was proposed. This is a progressive algorithm which performs multiple pairwise alignments, where the alignment from one pairwise comparison is used as one of the paths to be aligned in the next step. No automatic extraction of paths was performed as the paths were extracted from the pathway diagrams by "eye inspection", and no statistical significance assessment was carried out on the resulting alignments. A few experiments were performed on the KEGG pathways for metabolism of sugar, DNA and amino acids, where the multiple alignment algorithm was applied. However, only similar pathways were chosen for comparison, so the resulting alignments are not surprising.

Pinter et al. (2005) propose a metabolic pathway alignment method referred to as approximate labeled sub-tree homeomorphism, where the general idea is to search for matches by aligning a query pathway to a set of model pathways. Ranking of hits is done using a similarity score encompassing both the topological and semantic properties of the alignment. Metabolic enzyme-to-enzyme pathways were used, in which two enzymes are connected by a directed edge if the product of the reaction that the first enzyme catalyses is the substrate to the reaction that the second enzyme catalyses. As a background, graph homeomorphism is similar to graph isomorphism, where the latter occurs when a graph G_1 has a bijective mapping to another graph G_2 so that all node labels and edges of G_1 are preserved. Subgraph isomorphism is when G_1 is isomorphic to a subgraph of G_2. Subgraph homeomorphism occurs when a subdivision of G_1 is isomorphic to a subdivision of a subgraph in G_2. A subdivision of a graph is the graph obtained from the division of edges by node insertion. It is also possible to perform so called smoothing on an edge by removing a node between two other nodes and replacing it with a new edge. Subtree homeomorphism is a special case of subgraph homeomorphism only involving trees. For the approximate labeled subtree homeomorphism problem there is a scoring table for comparing the labels of the homeomorphic subtrees under study. This table contains the score contributions of individual node label comparisons. Since the scoring is approximate, the labels do not need to be identical for a score contribution. The total score is calculated by summing up all score contributions and also deducting the difference in sizes between the homeomorphic subtrees multiplied by a fixed node deletion (gap) penalty. Node deletion is inherent in the subtree homeomorphism concept. In Pinter et al. (2005), the scoring table contains the values of $-log_2(C(h))$, where h is the most specific enzyme nomenclature classification for enzymes E_i and E_j, and $C(h)$ denotes the number of enzymes whose classes are the same as or located under h in the hierarchy of the enzyme nomenclature. In the algorithm, both query and model pathways are converted into trees and a dynamic programming approach is used, exhibiting a time complexity of $O(m^2 \cdot n/log(m) + m \cdot n \cdot log(n))$, where m and n are the number of nodes in the subtrees under comparison. The statistical significance of alignments was assessed using the approach proposed by Maslov and Sneppen (2002), involving alignment of

the query pathway against 100 randomised versions of the model pathway. A p-value threshold of 0.01 was used. Both inter-species and intra-species comparison experiments were performed. In the inter-species case, which aims to find conserved metabolic pathways, all possible alignments were compared for the complete set of 113 *E. coli* pathways from EcoCyc and the 151 *S. cerevisiae* pathways from SGD. Only pathways containing at least two reactions were used. 610 pathway pairs out of 17063 had at least one significant alignment. Significant alignments were found for 78% of the 80 analogous pathways, which indicates that there is a high degree of evolutionary conservation between the species. Examples of analogous hits are the 17 enzyme pathway "phenyl-alanine, tyrosine and tryptophane biosynthesis" and the three enzyme pathway "homoserine to methionine biosynthesis". For the first pathway it appears that some enzymes are not identical by label or protein sequence similarity, but by enzyme nomenclature classification, demonstrating the utility of approximate semantic matching. In the second pathway, there was a gap in the *S. cerevisiae* subgraph, and the authors hypothesise that this may be due to gene fusion in *S. cerevisiae* or gene duplication in *E. coli*. The intra-species comparisons, aiming to track the metabolic evolution within a species, was performed by aligning all pathways against all for each species separately. For *E. coli* there were 187 (out of 12769) significant hits and for *S. cerevisiae* the corresponding figure was 262 (out of 22801). It was found that the biosynthesis pathways for valine and isoleucine are identical for both species, even the enzymes are exactly the same. This may imply that there is a single ancestor pathway from which these pathways have evolved.

An improvement to the approach for metabolic pathway alignment by Pinter et al. (2005) was presented by Wernicke and Rasche (2007). The authors argue that since the subgraph isomorphism (and homeomorphism) problem is NP-complete, earlier approaches to pathway alignment have relied on simplification of the pathway structure to make the pathway alignment algorithm feasible. Examples being choosing linear structures (Tohsato et al. 2000, Kelley et al. 2004) or trees (Pinter et al. 2005). The new method is not restricted to trees only and is faster than the Pinter method. Pathways are represented as connected directed graphs, where nodes are metabolites and edges are reactions. The alignment method is similar to Pinter et al. (2005) in

that subgraph homeomorphism is employed, but is referred to as embedding. The problem is to find the maximum score embedding of a pattern (query) graph G_P in a host (model) graph G_H. In the algorithm, for each vertex in G_H, a small partial embedding of G_P into G_H is calculated. Subsequently this embedding is extended in all possible ways using a recursive backtracking search in order to find the maximum-score embedding. For scoring, the same semantic scoring scheme as in Tohsato et al. (2000) was used, i.e. involving the enzyme nomenclature hierarchy. A linear gap penalty was also part of the scoring scheme. Unfortunately, exploring all possible extensions of the initial partial embeddings is very time consuming. A solution is presented where the concept of local diversity is introduced. The local diversity of metabolic networks refers to the observation that paths of reactions having the origin at the same metabolite (vertex) perform quite different functions, i.e. the enzyme classifications are very different. By using this observation, it is possible to avoid exploring certain solutions in the backtracking algorithm which do not appear to be biologically plausible. This change makes the algorithm for pathway alignment very efficient. The statistical significance of derived alignments is not assessed. The authors evaluated the method using 145 pathways for *B. subtilis*, 220 for *E. coli*, 190 for *H. sapiens*, 176 for *S. cerevisiae*, and 267 pathways for *T. thermophilus*. The very common metabolites ATP, ADP and H_2O were removed. All pathways of all species were compared against each other, and it is claimed that the resulting 996004 pathway comparisons only take 70 seconds. A few alignment examples are presented, and these are used to discuss the utility of the method. A few alignments were mentioned which were found by the method but not by the method in Pinter et al. (2005).

4.2 Method

The GOSAP method is summarised in figure 4.1. As input from the user, in the current implementation of the method, is given a model pathway graph and a query pathway graph. These are currently interchangeable, but are being distinguished by name since the intention for future versions of the implementation is to offer the user a choice of built-in model pathway graphs. In the current version of GOSAP, graphs

are assumed to be directed. The method involves three main procedures: 1) GO term probability calculation, 2) path extraction and 3) path alignment. The first procedure calculates the probability of the GO terms using an annotation database for one or several organisms. This knowledge is later used in the alignment procedure to calculate how semantically similar two gene products are. The second procedure systematically extracts paths from the model- and query pathway graphs. A path is here defined as a sequence of gene products appearing in succession in the pathway graph, when following the graph structure from a given node to a leaf node, without including any cycles. This procedure enables the alignment algorithm to handle graph structures by disassembling them into sets of paths which can be aligned individually. The path alignment procedure is where each of the query paths is aligned with each of the model paths. This is done using a modified version of the Smith-Waterman (Smith and Waterman 1981) local alignment algorithm, tailored for the alphabet of gene product identifiers and using a GO semantic similarity match function. Furthermore, a test for statistical significance of alignments is performed. More details regarding the three procedures are given in the following.

4.2.1 GO term probability calculation

Gene Ontology (Ashburner et al. 2000) is a structured vocabulary of molecular biology. It contains three different sub-ontologies covering the molecular functions, biological processes and cellular components of gene products, and is structured as a directed acyclic graph showing how terms are related to each other, using inheritance (IS-A) and aggregation (PART-OF). Using these relations, abstraction hierarchies of terms with different specificity are created. A gene product can be associated with several terms in each sub-ontology. One important use of GO is the calculation of semantic similarity between two terms. Here, an annotation database D is used to calculate the probability of each GO term using the procedure proposed by Lord et al. (2003a), which was introduced in section 2.4.1 and is reproduced here for clarity and convenience:

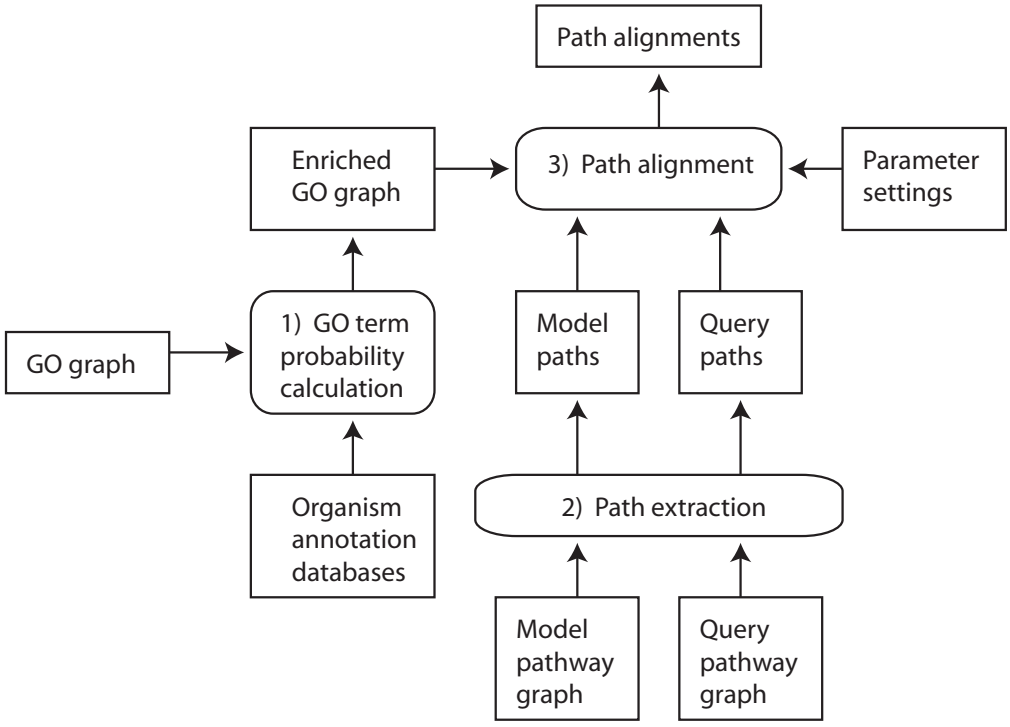


Figure 4.1: The GOSAP method. Boxes with rounded corners represent procedures, and rectangular boxes represent information.

For each gene product G_i in D:

Increment a counter C_j for each GO term T_j appearing in the annotation of G_i, and increment the counter of each ancestor term of T_j.

For each GO term T_k:

Calculate the term probability $p(T_k) = \frac{C_k}{N}$, where N is total number of annotations in D.

This is a well established procedure for calculation of concept probabilities in abstraction hierarchies or ontologies, and was also used by Resnik (1999) in his work on semantic similarity calculations using the WordNet (Miller 1990) vocabulary of the English language. Like in Lord et al. (2003a), both inheritance (is-a) and aggregation (part-of) relation types are considered in the term probability calculation. Terms of all evidence types are used. An evidence type indicates the kind of evidence used for assigning a particular GO annotation term to a gene product. An example is "traceable author statement" (TAS), which indicates that the information the choice of annotation term was based on was found in a published article or book.

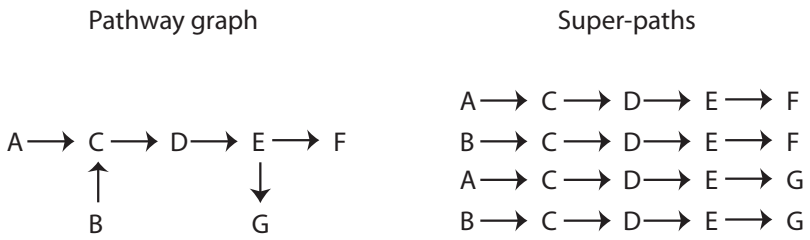


Figure 4.2: Decomposition of an example pathway graph into a complete set of super-paths.

4.2.2 Path extraction

An algorithm involving depth-first search was developed to derive all paths originating from each node in the pathway graph. Extension of a path ends whenever a leaf node or a previously visited node is encountered (so that cycles are handled). Furthermore, only the *super-paths* are used in the subsequent path alignment, i.e. the set of paths where no path is included in its entirety as a sub-path of another path. The purpose is to obtain a minimal set of paths, while still covering the entire pathway graph. To illustrate this principle, a schematic example pathway graph which is decomposed into a complete set of super-paths is shown in figure 4.2.

4.2.3 Path alignment

The well-known Smith-Waterman algorithm, which was originally developed for identification of common molecular subsequences (Smith and Waterman 1981), was here adapted for the task of producing local alignments of paths of gene products. A local alignment algorithm was chosen to ensure that the produced alignment can use path fragments rather than the full path as for global alignment algorithms. The scoring function s_f used for a match is defined in equation 4.1 (similar to the equation in Lord et al. (2003a)) and in equation 4.2 (proposed by Resnik (1999), introduced in this thesis in equation 2.1, but reproduced here as equation 4.2 for clarity and convenience).

$$s_f(G_i, G_j) = max(\{SS(T_k, T_l) : T_k \in t(G_i), T_l \in t(G_j)\}) \tag{4.1}$$

$$SS(T_k, T_l) = -log_2(p_{ms}(T_k, T_l)) \tag{4.2}$$

G_x refers to a gene product, $t(G_x)$ is the set of GO annotations for G_x, $SS(T_k, T_l)$ is the semantic similarity between GO molecular function terms T_k and T_l, and $p_{ms}(T_k, T_l)$ is the probability of the minimum subsumer of T_k and T_l. The minimum subsumer refers to the ancestor term with lowest probability that is common to both terms. We argue that the maximum value of all term comparisons for a gene product pair (G_i, G_j) in equation 4.1 is more appropriate for our purposes than the average value that was used in Lord et al. (2003a), because we are interested in finding the single most specific minimum subsumer to be included in the alignment (see section 4.3 for examples).

The biological process (P) and cellular component (C) sub-ontologies are used (in combination with the molecular function (F) ontology) to post-evaluate an alignment by calculating overall alignment scores:

$$s_{fp}(G_i, G_j) = s_f(G_i, G_j) + s_p(G_i, G_j) \tag{4.3}$$

$$s_{fpc}(G_i, G_j) = s_f(G_i, G_j) + s_p(G_i, G_j) + s_c(G_i, G_j) \tag{4.4}$$

The total alignment scores, denoted as S_F, S_{FP} and S_{FPC}, are calculated by summing up gene product comparison scores s_f, s_{fp}, s_{fpc} from equations 4.1, 4.3 and 4.4, and possible gap penalties, over all positions in the locally aligned segment. Hence, the scores s_{fp} and s_{fpc} enhance the "resolution" of similarity between paths, providing differentiation between alignments sharing similarity with respect to several sub-ontologies in combination.

As seen from equations 4.1, 4.3 and 4.4, the F sub-ontology was chosen for the purpose of guiding the alignment procedure, whereas the P and C sub-ontologies are only used for post-evaluation of alignments. This ensures that the alignment of pairs of gene products that share similar function is promoted. If, for example, the P sub-ontology had been used to guide the alignment process, it would simply promote alignments where the pairs of gene products appearing in a single position of the alignment belong to similar processes, without enforcing this process similarity throughout the length of the alignment.

When deriving an alignment using nucleotides or amino acids, it is better to start a new alignment when the best dynamic programming matrix option is 0, motivated

by the assumption that scores of random matches are negative. In GOSAP, a random match is expected to have a value of 0.6, which is the average term-to-term semantic similarity according to equation 4.2 when all pairs of molecular function terms in GO are compared. Furthermore, a linear gap penalty is used in the local alignment algorithm.

Statistical significance of alignments

The alignment score itself may not be sufficient for judging the quality of an alignment. Therefore, an assessment of the statistical significance of alignments was performed according to the procedure described by Maslov and Sneppen (2002). In this procedure two edges $A \rightarrow B$ and $C \rightarrow D$ are randomly selected in a graph and rewired into $A \rightarrow D$ and $C \rightarrow B$. If the resulting edges are already present in the graph, a new pair of edges is selected. Hence, a randomisation takes place while preserving the cardinality of each node (gene product). A series of random edge switches results in a randomised graph, with the restriction that the randomised graph must be different from the original graph.

The procedure in Maslov and Sneppen (2002) does not work on small graphs or graphs with a central node connecting to a number of neighbouring nodes (e.g. $A \rightarrow B$, $A \rightarrow C$ and $A \rightarrow D$), since no randomisation is possible. This is the case for a number of graphs in our metabolic pathway experiments. Therefore, unlike for the initial experiments on regulatory pathways, we instead used *node shuffle* for the experiments on metabolic pathways. Node shuffle is an alternative procedure for graph randomisation where node identifiers of a graph are simply randomly shuffled while keeping the edges. The resulting graph has the same number of edges and the same topology, but the cardinality of individual nodes is not preserved. The node shuffle procedure can handle any graph with two or more nodes and one or more edges. As in the Maslov and Sneppen procedure, the randomised graph is required to be different from the original graph.

In GOSAP, a query path can be aligned with a large number of randomised versions of the model pathway using the Maslov and Sneppen procedure for larger graphs or the node shuffle procedure for smaller graphs. The p-value of an alignment is defined

as the fraction of randomised pathways that produce an alignment with equal or higher score than the original alignment. Low p-values are desirable.

4.3 Results

We have tested our algorithm on protein regulatory pathways as well as metabolic enzyme-to-enzyme pathways in order to demonstrate that it is useful.

4.3.1 On the complexity

This subsection elaborates on the time complexity of the GOSAP method. Referring to the method description in the previous section, the complexity of the GO term probability calculation is the same as for the GOTEM method, which is described in section 3.3.1. The path extraction has a complexity of $O(2|V| \cdot b^m)$, where V is the set of vertices in the model- or query graph, b is the branching factor of a vertex, and m is the maximum depth of a path in the model- or query graph.

The path alignment has a complexity of $O(|P_m| \cdot |P_q| \cdot [|G_m| \cdot |G_q| \cdot |GT_m| \cdot |GT_q| + |GT_x|])$, where P_m is the set of model paths, P_q is the set of query paths, G_m is the set of gene products in a path from P_m, G_q is the set of gene products in a path from P_q, GT_m denotes the set of GO terms with ascendants for a gene product in G_m, GT_q is the set of GO terms with ascendants for a gene product in G_q, and $|GT_x|$ is the number of gene products traversed until a local optimum alignment has been assembled. Hence, the inner loop within square brackets refers to the complexity of the Smith-Waterman algorithm and its matrix fill and alignment derivation procedures.

The complexity of the statistical significance calculations is N_r times the path alignment complexity above, since each query path is aligned against all paths in the N_r randomised versions of the model graph. The randomised model graphs are derived prior to the path alignment and statistical significance calculations, so that the same set of randomised graphs is used for every query path. The Maslov and Sneppen algorithm for graph randomisation has linear complexity with execution time proportional to the number of edge switches. With larger graphs, a larger number of edge switches is required in order to perform a proper randomisation. The node

shuffle procedure can be implemented with linear complexity $O(|V|)$, where V is the set of vertices in the model- or query graph, by using the algorithm proposed by Durstenfeld (1964) to generate a random permutation out of a finite set of objects.

4.3.2 Protein regulatory pathways

In the first experiment, protein regulatory pathways from KEGG (Kanehisa and Goto 2000) were used. When a gene product complex regulates another complex, a link was created from each gene product in the regulating complex to each gene product in the regulated complex. The cell cycle regulatory pathway for *S. cerevisiae* was used as model pathway and the MAPK signaling pathway was used as query. As these two pathways are quite different, few similarities are expected to be detected, and this initial experiment is mainly presented to illustrate the method and the properties of the derived alignments. In the path extraction pre-processing step, 195 super-paths were extracted from the *S. cerevisiae* cell cycle pathway and 37 from the MAPK pathway. Using a gap penalty of 1, 100 path randomisations for the significance tests, the Maslov and Sneppen randomisation procedure, and the statistical significance threshold $p \leq 0.05$, six alignments in total were found for 3 of the 37 paths extracted from the MAPK pathway graph. For example, the query path "MSG5 $\xrightarrow{i}$ FUS3 $\xrightarrow{p}$ FAR1", resulted in the following alignment with $S_F = 25.8$ and the significance value $p_f = 0.04$:

Alignment							
Q:	MSG5	$\xrightarrow{i}$	FUS3		*gap*		FAR1
M:	MIH1	$\xrightarrow{d}$	CDC28	$\xrightarrow{ip}$	SWI5	$\xrightarrow{e}$	SIC1
Meta-alignment							
F:	0004725	$\rightarrow$	0004674		*gap*		0019210
P:	0050875	$\rightarrow$	0006468		*gap*		0000074
C:	0005737	$\rightarrow$	0005634		*gap*		0005634

Q and M are the aligned paths for query and model, respectively. F shows the GO molecular function meta-alignment, where each identifier represents the minimum subsumer GO term for the two gene products under comparison. The corresponding

information for biological process and cellular component is shown in the rows labelled by P and C. Symbols above arrows denote relation types as specified in KEGG; "i"=inhibition, "d"=dephosphorylation, "ip"=inhibition and phosphorylation, and "e"=expression. For example, in the molecular function meta-alignment, gene products MSG5 and MIH1 have the minimum subsumer "protein tyrosine phosphatase activity" (GO:0004725). This can also be observed in figure 4.3, which illustrates the molecular function GO annotation of the gene products appearing in the alignment. Figure 4.3 also shows how the GO terms are placed relative to each other in the term hierarchy.

The identity score, which represents a perfect match, is defined as the semantic score obtained when comparing the ungapped query alignment segment with itself. It is 36.4 for the query "MSG5 $\xrightarrow{i}$ FUS3 $\xrightarrow{p}$ FAR1", and the obtained $S_F = 24.8$ is therefore 71% of the identity score. Another significant alignment for the same query path was obtained using a combined score where "FUS3 $\xrightarrow{p}$ FAR1" was aligned with exactly the same sub-sequence ("FUS3 $\xrightarrow{p}$ FAR1") in the model with $S_{FPC} = 53.3$ and the significance value $p_{fpc} = 0.04$, which demonstrates the utility of using all three ontologies to score alignments. This path is present in both the MAPK and cell cycle pathways in KEGG, because it is both an exit point from the MAPK pathway and an entry point to the cell cycle pathway. (For explanations of all GO identifiers appearing in the example alignments, the reader is referred to the Gene Ontology website at www.geneontology.org).

4.3.3 Reverse engineered regulatory pathways

Apart from studying similarities among documented pathways, our method can be used to assess hypothetical regulatory pathways derived using reverse engineering techniques. This makes it possible to evaluate the hypothetical pathways that are derived from experimental data by comparison to known pathways. Since the comparison is based on general molecular functions, encoded by the GO terms, the identified similarities with known pathways can be interpreted as indications of biological plausibility of the derived hypothetical pathways.

To test this possible application scenario for the GOSAP method, we used a pu-

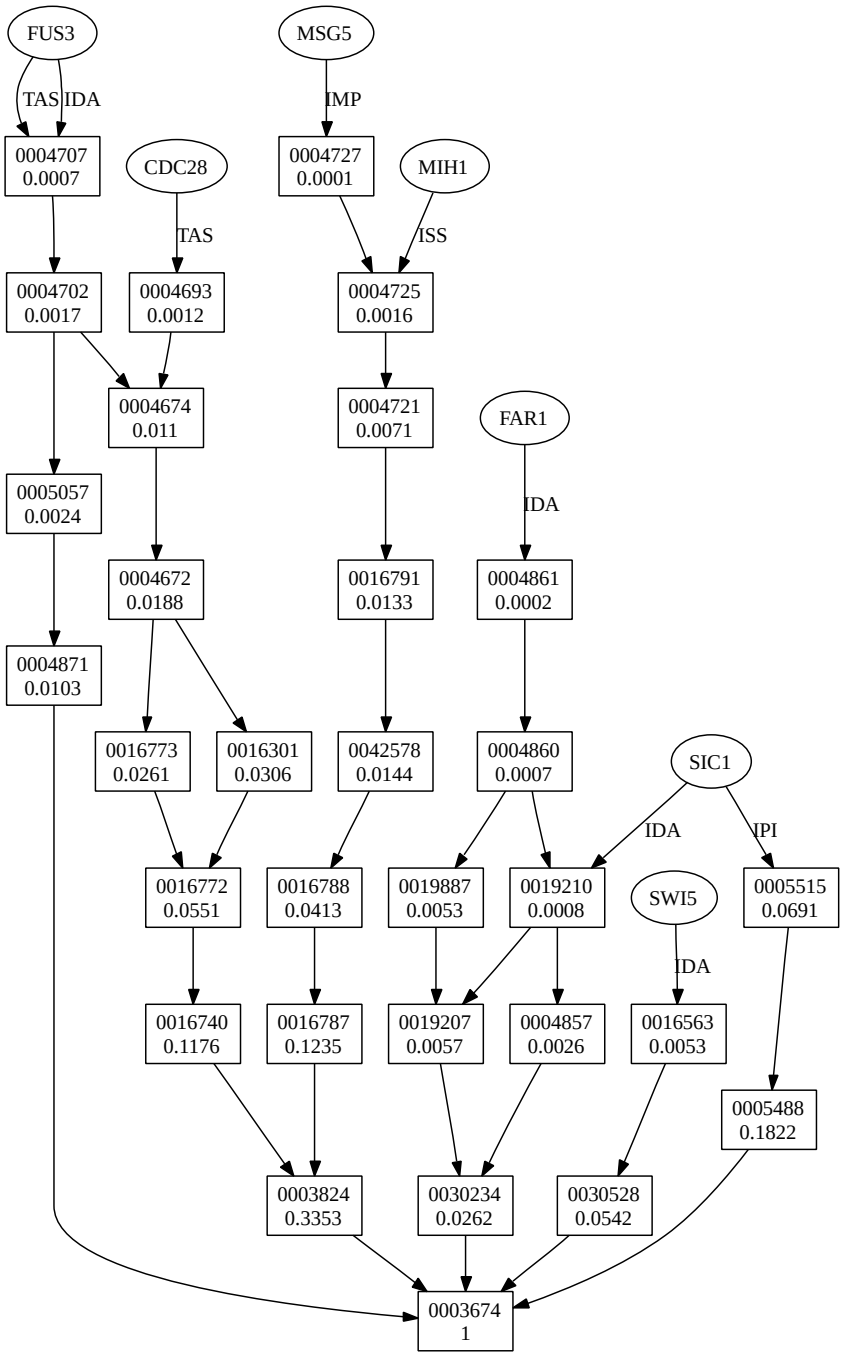


Figure 4.3: Gene products (ovals) mapped to GO terms (rectangles) according to their molecular function annotation. "0003674" represents the most abstract GO term "molecular function" which has a probability of 1 (number in lower part of rectangles). Descendants of this term have lower probabilities and are therefore more specific.

tative pathway reported by Kim et al. (2003) containing 12 gene products and 14 edges, which had been derived with a dynamic Bayesian network technique, using microarray gene expression data for a subset of the *S. cerevisiae* cell cycle as input. When manually comparing the derived pathway model to the KEGG regulatory pathway model of the *S. cerevisiae* cell cycle, it was observed that only 3 of 14 edges in this derived pathway are identical to edges occuring in the KEGG pathway. Using GOSAP, the 11 super-paths that can be extracted from the derived pathway model were aligned with 195 super-paths extracted from the KEGG model. Using a gap penalty of 1, 100 randomised models, and statistical significance threshold $p \leq 0.05$, one significant alignment was found for the query path "FAR1 $\stackrel{?}{\rightarrow}$ SIC1 $\stackrel{?}{\rightarrow}$ CLN2 $\stackrel{?}{\rightarrow}$ SIC1" with $S_{FP} = 76.6$ and a significance value $p_{fp} = 0.03$:

Alignment										
Q:	FAR1	$\stackrel{?}{\rightarrow}$	SIC1		*gap*		CLN2	$\stackrel{?}{\rightarrow}$	SIC1	
M:	FAR1	$\stackrel{i}{\rightarrow}$	CLN1	$\stackrel{p}{\rightarrow}$	SWI6	$\stackrel{e}{\rightarrow}$	CLN2	$\stackrel{p}{\rightarrow}$	SIC1	
Meta-alignment										
F:	0004861	$\rightarrow$	0019207		*gap*		0016538	$\rightarrow$	0019210	
P:	$\left\{\begin{array}{l}0007050\\0045786\end{array}\right\}$	$\rightarrow$	0000079		*gap*		$\left\{\begin{array}{l}0000320\\000321\end{array}\right\}$	$\rightarrow$	0000079	
C:	0005634	$\rightarrow$	0005634		*gap*		0005634	$\rightarrow$	0005634	

The first and fourth position in the process meta-alignment have two terms as equally good alternatives with respect to score. The meta-alignments show for example that SIC1 and CLN1 have the molecular function "kinase regulator activity" (GO:0019207), the biological process "regulation of cyclin dependent protein kinase activity" (GO: 0000079) and the cellular component "nucleus" (GO:0005634), as minimum subsumers. The correct sub-path "CLN2 $\stackrel{p}{\rightarrow}$ SIC1" in the derived model is captured, and the gap in the query sequence is aligned with the transcription co-activator SWI6 in the model. This suggests that SWI6 is a step that is missing in the corresponding hypothetical pathway, which gives us important clues to how the derived pathway can be modified to better reflect the current knowledge. Additionally, clues to potential relation types for the query path are provided by the model path, e.g. a phosphorylation relation is feasible between CLN2 and SIC1. Once again, the utility of combined scores is demonstrated, since no significant alignment could be found using the molec-

ular function score alone ($p_f \geq 0.36$). An identical significant alignment was found when using all three sub-ontologies.

4.3.4 Metabolic pathways

GOSAP was also used to compare the metabolic pathways of *S. cerevisiae* and *E. coli*, which are documented in the SGD (Saccharomyces Genome Database, www.yeastgenome.org). Metabolic enzyme-to-enzyme pathways were used, which are derived from ordinary metabolic pathways by creating directed links between enzymes if the product of one enzymatic reaction is the substrate of another enzymatic reaction (figure 4.4). This procedure has been used by others (see for example Pinter et al. (2005)). The SGD GO term annotations found at the GO website (www.geneontology.org) were used for *S. cerevisiae*. Since no annotations are available for *E. coli* at this website, the EBI GO annotations (www.ebi.ac.uk) were used for *E. coli*. It was found that a large proportion of the *E. coli* gene products lack cellular component annotation. Therefore, two separate sets of pathways were created; one set of 125 pathways where all gene products have annotations for molecular function and biological process but lack cellular component, and one set of 33 pathways where annotations from all three sub-ontologies are available. For the 103 *S. cerevisiae* pathways, all gene products had annotations from all sub-ontologies.

Figure 4.5 illustrates the distribution of number of nodes, edges and paths for the three different pathway sets used in the experiments. It can be observed that metabolic pathways are in general quite small as approximately 80% of the pathways in each pathway set have 10 nodes or less (86 out of 103 pathways for *S. cerevisiae*, and a greater share for the other two pathway sets). The distribution of edges is very similar to the node distribution, showing that the connectivity of nodes is limited. This is not surprising, since metabolic pathways are in most cases linear chains of reactions. The number of super-paths is also limited, as a result of the small number of nodes and limited connectivity of metabolic networks. It can be observed that more than 50% of the pathways in each pathway set has only one or two super-paths (58 out of 103 pathways for *S. cerevisiae*, and a greater share for the other two pathway sets). Table 4.1 shows some statistics on the pathway sets.

Table 4.1: Pathway statistics. SC-FPC is the *S. cerevisiae* pathway set where gene products are annotated with all sub-ontologies of GO, EC-FPC is the corresponding pathway set for *E. coli*, and EC-FP is the *E. coli* pathway set where GO cellular component is missing. For the number of nodes (N), edges (E) and paths (P) are stated the average, standard deviation, minimum and maximum.

	SC-FPC	**EC-FPC**	**EC-FP**
avg(N)	6.5	3.7	6.5
std(N)	4.8	2.5	4.9
min(N)	2	2	1
max(N)	30	11	31
avg(E)	6.8	3.1	6.6
std(E)	6.8	3.4	7.0
min(E)	1	1	1
max(E)	41	16	41
avg(P)	6.3	2.5	5.0
std(P)	11.9	3.2	8.8
min(P)	1	1	1
max(P)	92	18	56

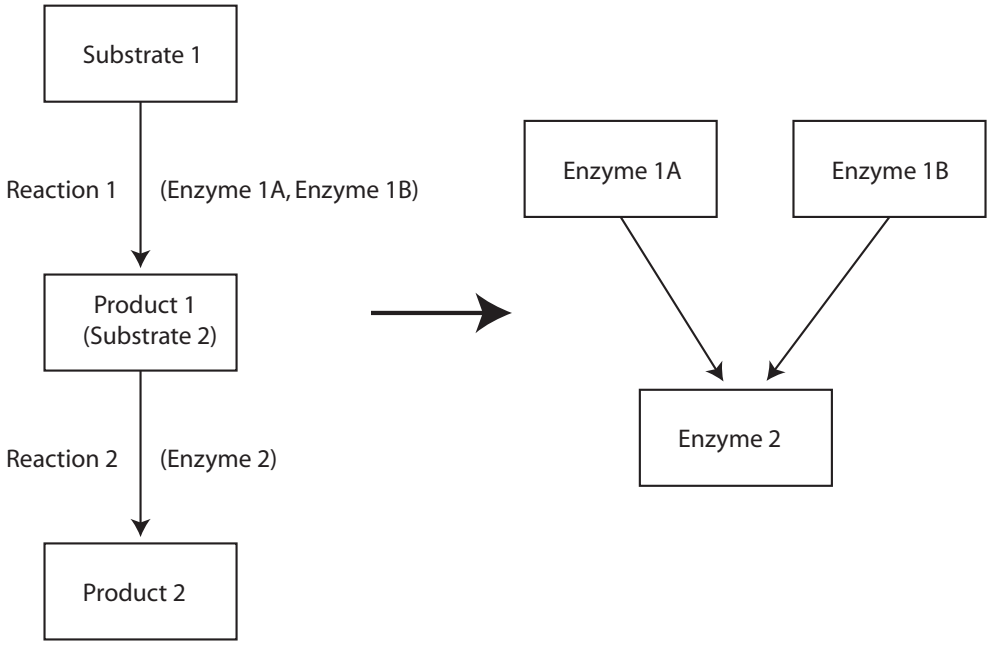


Figure 4.4: Conversion of metabolic pathway to enzyme-to-enzyme pathway. If the product of reaction 1 is the same as the substrate of reaction 2, the enzymes of reaction 1 are connected to the enzymes of reaction 2.

Pinter et al. (2005) performed an experiment where all pathways in one pathway set were aligned with all pathways in another pathway set. In their work, a significant pathway comparison occured if there was at least one significant alignment hit with $p \leq 0.01$ when comparing two pathways. The exact binomial test (Conover 1971) was performed in order to investigate if the number of observed pathway comparisons deviated significantly from the number of pathway pairs expected by chance. This test calculates the probability of k successes (number of significantly aligned pathway pairs) for n Bernoulli experiments (total number of pathway comparisons) using a probability p of success, where p is the significance threshold used in the alignment algorithm. Here, we perform the same analysis and extend it to investigate different thresholds of p for the investigated pathway sets. The R software (www.r-project.org) was used for the exact binomial test, just as in Pinter et al. (2005). Table 4.2 shows the number of significantly aligned pathway pairs as a function of threshold values for p and alignment length, and this data is used to perform the test. As an example, the number of alignments expected by chance in the comparison between pathways

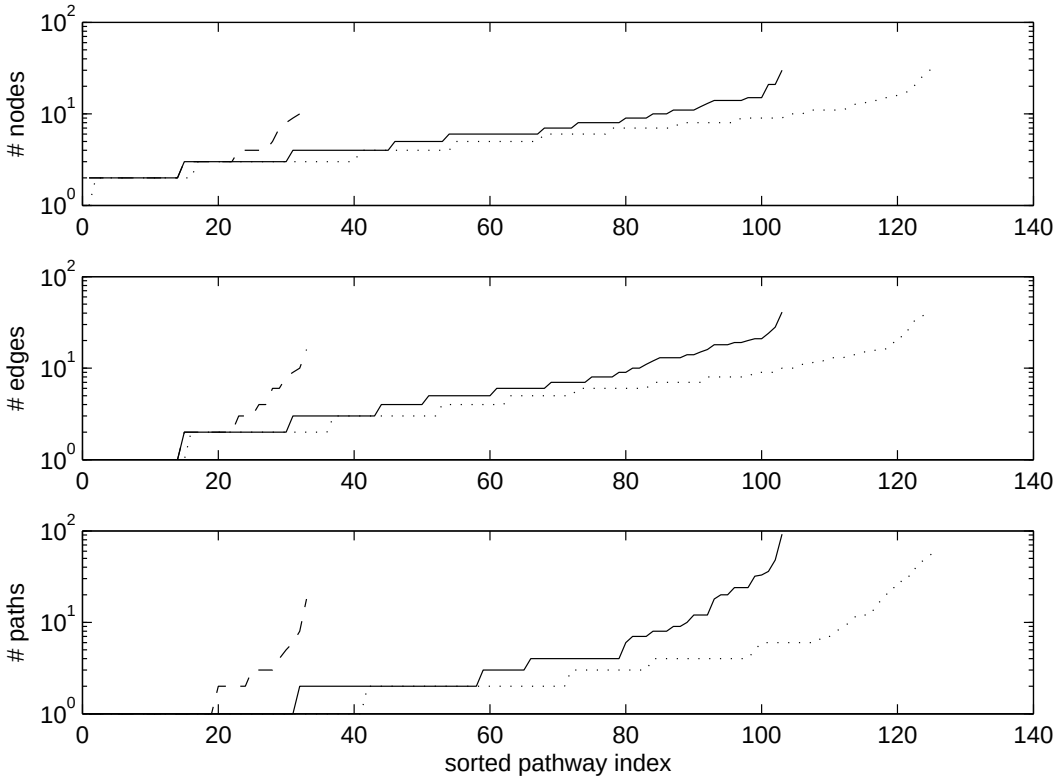


Figure 4.5: Pathway distribution diagrams showing number of nodes, edges and paths as a function of sorted pathway index. Solid lines represent the set of 103 pathways for *S. cerevisiae* where annotations for all sub-ontologies are used, dotted lines represent the 125 pathways for *E. coli* using the F and P sub-ontologies, and dashed lines are used for the *E. coli* set of 33 pathways where annotations from all sub-ontologies are used.

for *S. cerevisiae* and *E. coli* using $p \leq 0.02$ would be $103 \cdot 125 \cdot 0.02 = 258$. Using the exact binomial test, we can derive the probability p_b that this number differs significantly from those in table 4.2, i.e. from 426 and 474 in this example. For all cases where $p = 0$ the binomial test yields $p_b \leq 2 \cdot 10^{-16}$. With $p \leq 0.02$ and $l \geq 2$ the binomial test gives $p_b \leq 2 \cdot 10^{-16}$ in all cases except for the *S. cerevisiae* intra-pathway comparison where $p_b = 3.5 \cdot 10^{-15}$. Calculations for other alignment length thresholds is not considered relevant since $l = 2$ is the minimum length of an alignment, and the other alignment lengths in table 4.2 are only used in order to show how the number of alignments varies as a function of length. Results when $p \leq 0.04$ and $l \geq 2$ show that $p_b = 0.17$ in two cases (the SE comparison using the F sub-ontology, and the SS comparison using both F and P sub-ontologies), and $p_b \leq 0.04$ in all other cases.

Experiments were performed that are similar to those in Pinter et al. (2005), where the metabolic enzyme-to-enzyme pathways were extracted and compared for the same organisms. A gap penalty of 1 was chosen, and 100 randomised models were used in the statistical significance test. The *node shuffle* procedure for model randomisation was applied. Table 4.3 shows semantic relations between different pathways for *E. coli* and *S. cerevisiae* that were found using $p \leq 0.04$, path length ≥ 3 and score quota $sc/sc_{id} \geq 0.7$. A relation between two pathways is considered to be detected if at least one significant alignment is found between super-paths derived from the two pathways being compared. A total of 60 such relations, i.e. pathway pairs, were found, where 38 can be identified by their names as being functionally analogous between the two organisms. It was observed that sets of pathway relations in table 4.3 form isolated sub-graphs.

A set of relations was identified between biosynthesis pathways for phenylalanine, tyrosine and tryptophan, which was also identified in Pinter et al. (2005). These relations have the following numbers in table 4.3: {4-5,39-40,54-57} (sub-graph A in figure 4.6). This finding is supported by the theory that amino acid biosynthesis pathways were established prior to the divergence into the kingdoms Archaea, Bacteria and Eukaria (Hochuli et al. 1999). Additionally, GOSAP identified a relation not reported in Pinter et al. (2005) between *E. coli* chorismate biosynthesis and the combined *S. cerevisiae* biosynthesis pathway for phenylalanine, tyrosine and tryptophan.

Table 4.2: Number of significantly aligned pathway pairs as a function of thresholds for p and alignment length (l). *SE* refers to the comparison between *S. cerevisiae* and *E. coli* pathways, *SS* denotes *S. cerevisiae* intra-pathway comparison and *EE* is the corresponding intra-pathway comparison for *E. coli*. The three figures in the *SS* column separated by dash (/) signs represent the use of GO molecular function alone, molecular function combined with biological process, and all sub-ontologies, respectively. For the *SE* and *EE* comparisons, the alternative with all sub-ontologies was not used due to the poor cellular component annotation of *E. coli*, therefore only two figures are shown.

p	l	**SE**	**SS**	**EE**
0	2	355/369	281/329/354	925/642
0	4	102/137	118/138/155	299/288
0	6	40/52	56/66/74	75/69
0	8	17/18	25/33/42	29/25
0.02	2	426/474	335/391/436	1070/821
0.02	4	158/217	160/185/216	404/419
0.02	6	55/75	81/87/101	110/102
0.02	8	24/30	34/43/48	35/30
0.04	2	484/574	383/452/493	1222/957
0.04	4	197/294	197/233/259	518/522
0.04	6	74/105	96/109/119	143/120
0.04	8	28/41	40/56/57	45/33

Table 4.3: Semantic pathway relations for the GOSAP comparison between the metabolic enzyme-to-enzyme pathways in *E. coli* and *S. cerevisiae*. A relation between two pathways is here considered to occur if GOSAP reports at least one significant alignment ($p \leq 0.04, l \geq 3, sc/sc_{id} \geq 0.7$).

#	E. coli	S. cerevisiae
1	arginine biosynthesis I	arginine biosynthesis
2	biosynthesis of proto- and siroheme	heme biosynthesis
3	biotin biosynthesis I	biotin biosynthesis
4	chorismate biosynthesis	chorismate biosynthesis
5	chorismate biosynthesis	phenylalanine, tyrosine and tryptophan biosynthesis
6	colanic acid building blocks biosynthesis	colanic acid building blocks biosynthesis
7	colanic acid building blocks biosynthesis	lactose degradation
8	de novo biosynthesis of pyrimidine deoxyribonucleotides	de novo biosynthesis of pyrimidine deoxyribonucleotides
9	de novo biosynthesis of pyrimidine ribonucleotides	de novo biosynthesis of pyrimidine ribonucleotides
10	fatty acid elongation – saturated	fatty acid biosynthesis, initial steps
11	fatty acid elongation – unsaturated	fatty acid biosynthesis, initial steps
12	fatty acid oxidation pathway I	fatty acid oxidation pathway
13	galactose degradation I	lactose degradation
14	galactose degradation I	colanic acid building blocks biosynthesis
15	gluconeogenesis	gluconeogenesis
16	glycolysis I	glycolysis
17	glycolysis I	aerobic glycerol catabolism
18	glycolysis I	gluconeogenesis
19	glyoxylate cycle	serine-isocitrate lyase pathway
20	histidine biosynthesis I	histidine biosynthesis
21	homoserine and methionine biosynthesis	methionine biosynthesis
22	homoserine and methionine biosynthesis	threonine and methionine biosynthesis
23	homoserine biosynthesis	threonine and methionine biosynthesis
24	homoserine biosynthesis	homoserine biosynthesis
25	homoserine biosynthesis	methionine biosynthesis
26	homoserine biosynthesis	threonine biosynthesis
27	isoleucine biosynthesis I	valine biosynthesis
28	isoleucine biosynthesis I	isoleucine biosynthesis
29	leucine biosynthesis	leucine biosynthesis
30	methionine and methyl-donor-molecule biosynthesis	methionine biosynthesis
31	methionine and methyl-donor-molecule biosynthesis	homoserine biosynthesis
32	methionine and methyl-donor-molecule biosynthesis	threonine biosynthesis
33	methionine and methyl-donor-molecule biosynthesis	threonine and methionine biosynthesis
34	methionine biosynthesis I	sulfur amino acid biosynthesis
35	methylcitrate cycle	serine-isocitrate lyase pathway
36	NAD biosynthesis I (from aspartate)	de novo NAD biosynthesis
37	pantothenate biosynthesis I	pantothenate and coenzyme A biosynthesis
38	phenylacetate degradation I	fatty acid oxidation pathway
39	phenylalanine biosynthesis I	phenylalanine, tyrosine and tryptophan biosynthesis
40	phenylalanine biosynthesis I	phenylalanine biosynthesis
41	phospholipid biosynthesis I	phosphatidic acid and phospholipid biosynthesis
42	phospholipid biosynthesis I	phospholipid biosynthesis (Kennedy pathway)
43	purine nucleotides de novo biosynthesis I	de novo biosynthesis of purine nucleotides
44	pyridine nucleotide cycling	de novo NAD biosynthesis
45	respiration (anaerobic)	serine-isocitrate lyase pathway
46	riboflavin and FMN and FAD biosynthesis	riboflavin, FMN and FAD biosynthesis
47	salvage pathways of pyrimidine ribonucleotides	salvage pathways of pyrimidine ribonucleotides
48	sulfate assimilation	sulfate assimilation pathway II
49	sulfate assimilation	sulfur amino acid biosynthesis
50	tetrahydrofolate biosynthesis	folic acid biosynthesis
51	threonine biosynthesis	threonine biosynthesis
52	threonine biosynthesis	threonine and methionine biosynthesis
53	threonine biosynthesis	methionine biosynthesis
54	tryptophan biosynthesis	tryptophan biosynthesis
55	tryptophan biosynthesis	phenylalanine, tyrosine and tryptophan biosynthesis
56	tyrosine biosynthesis I	tyrosine biosynthesis
57	tyrosine biosynthesis I	phenylalanine, tyrosine and tryptophan biosynthesis
58	ubiquinone biosynthesis	ubiquinone biosynthesis
59	valine biosynthesis	isoleucine biosynthesis
60	valine biosynthesis	valine biosynthesis

Chorismate biosynthesis is a precursor to the later mentioned pathway.

Another interesting example is the relation set {21-26,30-33,51-53} which is shown in sub-graph B of figure 4.6, involving the biosynthesis pathways for the amino acids homoserine (a variant of serine), methionine and threonine. The relation between the biosynthesis pathways for homoserine and methionine was also found in Pinter et al. (2005) and this result also supports the hypothesis about a common ancestral biosynthesis pathway (Hochuli et al. 1999).

GOSAP also identified relations 27 and 59 between the biosynthesis pathways for valine and isoleucine, which were also reported in Pinter et al. (2005). This is a highly plausible finding, since it is believed that the biosynthesis of these pathways originates from some common ancestral biosynthesis pathway (Klipcan and Safro 2004). This relation is based on an ungapped alignment of length four which has the term "branched chain family amino acid biosynthesis" (GO:0009082) at all four positions in the process meta-alignment. This supports the hypothesis of a common biosynthesis pathway, as the children of this term are "isoleucine biosynthesis", "leucine biosynthesis" and "valine biosynthesis". This again demonstrates the utility of the P (process) sub-ontology.

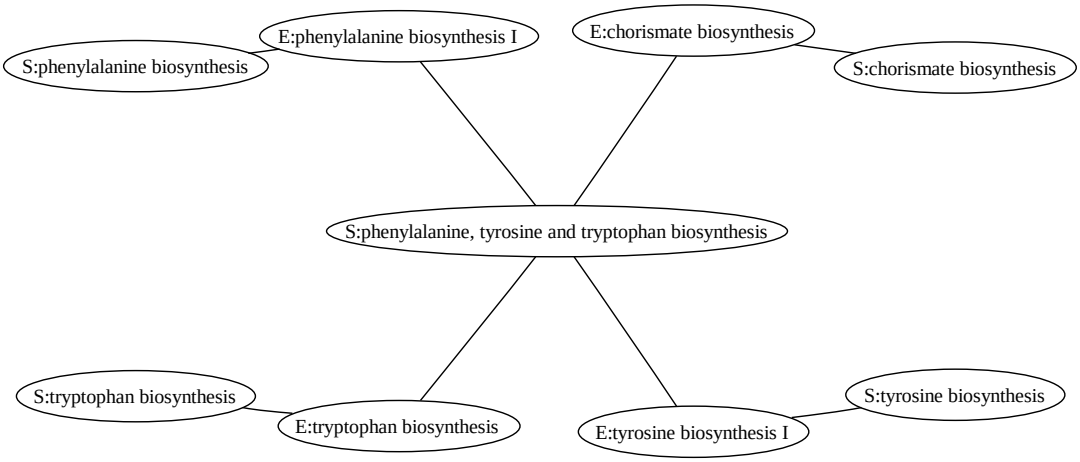
We also found semantic relation sub-graphs not reported in Pinter et al. (2005), which indicates a greater sensitivity of GOSAP. An example is the one in sub-graph C of figure 4.6, which involves the relation set {19,35,45}. The relation between the serine-isocitrate lyase pathway of *S. cerevisiae* and the glyoxylate cycle of *E. coli* is for example explained by an alignment of length four between semantically identical paths. This is not surprising since the metabolite glyoxylate is a part of the serine-isocitrate lyase pathway (glyoxylate + L-serine = L-glycine + hydroxypyruvate) according to SGD. The relation between "serine-isocitrate lyase pathway" and "methylcitrate cycle" of *E. coli* emerges due to a length four alignment with the molecular function meta-alignment "transferase activity, transferring acyl groups, acyl groups converted into alkyl on transfer" → "hydro-lyase activity" → "aconitate hydratase activity" → "oxo-acid-lyase activity". The relation between "serine-isocitrate lyase pathway" and "respiration (anaerobic)" of *E. coli* includes an alignment where the molecular function meta-alignment is "citrate (Si)-synthase activity" → "aconitate

hydratase activity" → "aconitate hydratase activity" → "catalytic activity".

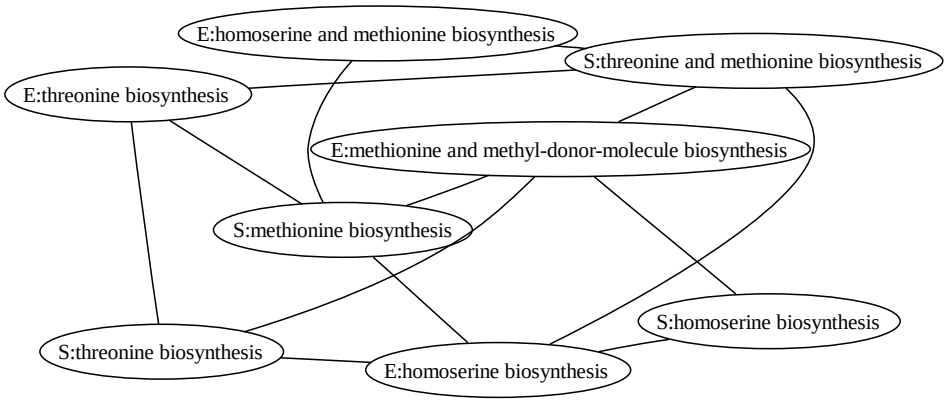
Another interesting sub-graph (figure 4.7) was found when the sc/sc_{id} threshold was increased to 0.9 and the path length threshold was decreased to 2. The sub-graph shows relations between similar sugar alcohol degradation pathways (Hexitol, Sorbitol, Mannitol and Galactitol) for the two organisms. These are considered as related because of an alignment between the path PFK2 → FBA1 of *S. cerevisiae* and the path K6PF1 → ALF1 of *E. coli*. PFK2 and K6PF1 are semantically identical with respect to molecular function since both are involved in "6-phosphofructokinase activity". FBA1 and ALF1 are also identical, since they share the molecular function "fructose-bisphosphate aldolase activity". The relation between "glycolysis I" of *E. coli* and the sugar alcohol degradation pathways of *S. cerevisiae* emerges because PFK2 → FBA1 is aligned with K6PF1 → ALF1, i.e. exactly the same alignment that identifies the relation between the sugar alcohol degradation pathways for the two organisms. The relation between "glycerol and glycerolphosphodiester degradation" for *E. coli* and "aerobic glycerol catabolism" for *S. cerevisiae* emerges because of a semantically perfect alignment with respect to molecular function between the paths GLPK → GLPA and GUT1 → GUT2. Common molecular functions for the first and second gene product are "glycerol kinase activity" and "glycerol-3-phosphate dehydrogenase activity". "Aerobic glycerol catabolism" of *S. cerevisiae* and "glycolysis I" of *E. coli* are related due to a semantically almost perfect and gap-free alignment involving five gene products, where the molecular function meta-alignment is "glyceraldehyde-3-phosphate dehydrogenase (phosphorylating) activity" → "phosphoglycerate kinase activity" → "phosphoglycerate mutase activity" → "phosphopyruvate hydratase activity" → "pyruvate kinase activity". The relation between "glycolysis" of *S. cerevisiae* and "glycolysis I" of *E. coli* has an alignment where the meta-alignment is identical to the aforementioned meta-alignment.

Using the same data set, we also investigated whether there is any correlation between the score quota $sc/scid$, p and length (l) of alignments. The results are shown in the upper part of table 4.4. All super-paths of all pathways in one pathway set were aligned with all super-paths in another pathway set, creating vectors of values for score quota, p and alignment length. The vectors were compared using standard

A)



B)



C)

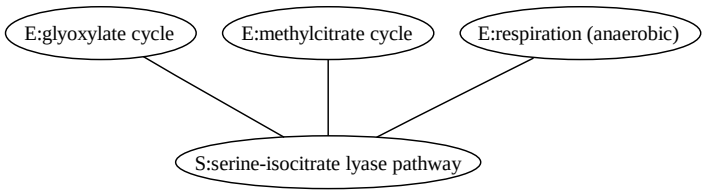


Figure 4.6: Semantic pathway relation sub-graphs derived using a GOSAP comparison between *S. cerevisiae* pathways (denoted with the prefix S:) and *E. coli* pathways (prefix E:). The requirement for a relation between two pathways is that at least one significant alignment was found by GOSAP, using thresholds $p \leq 0.04, l \geq 3, sc/sc_{id} \geq 0.7$.

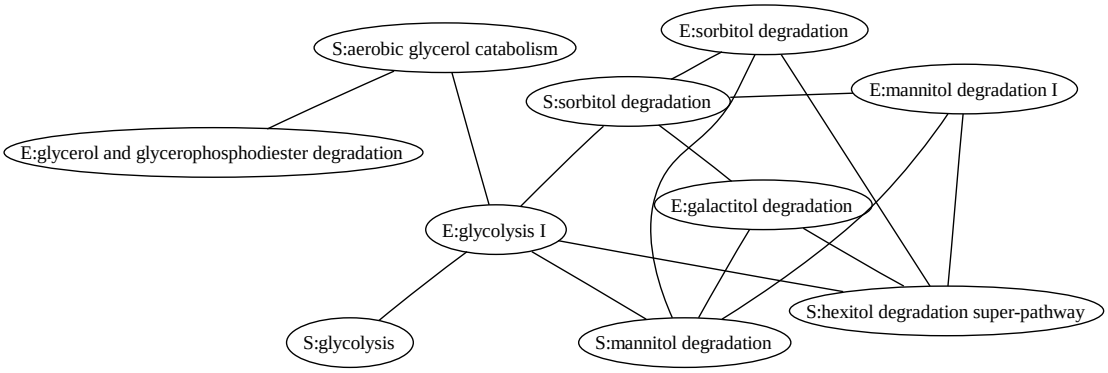


Figure 4.7: Semantic pathway relation sub-graph. Threshold settings: $p \leq 0.04$, $l \geq 2, sc/sc_{id} \geq 0.9$

Pearson correlation. It can be observed that there is a weak negative correlation between the score quota and p for most of the compared pathway sets. It is natural with a negative correlation since good alignments generally have a high score quota and a low p, whereas bad alignments usually have a low score quota and a high p. Score quota and alignment length seem rather uncorrelated, but there is a weak negative correlation between p and alignment length. This implies that low p-values are more likely when long alignments are obtained.

Additionally, we studied how the use of several sub-ontologies affects the p-value of alignments. This was done by dividing; A) the number of alignments where p for the larger set of sub-ontologies is smaller than the corresponding p using the smaller set of sub-ontologies, with; B) the number of alignments where p for the smaller set of sub-ontologies is smaller than the corresponding p using the larger set of sub-ontologies. The results are shown in the lower part of table 4.4. It is evident for all pathway set comparisons, except one, that lower p-values occur more frequently when using the P sub-ontology in combination with F instead of using F alone, since the quota is greater than one. The only exception is the *E. coli* intra-pathway comparison. One reason for this may be that different organisms have annotations of different quality. Adding the C sub-ontology also improves on the situation of using F alone. However, there seems to be a much smaller difference between using all sub-ontologies and the combination of F and P. In the case of the *S. cerevisiae* intra-pathway comparison,

Table 4.4: Path alignment statistics.

	SS (FPC)	**SE (FPC)**	**SE (FP)**	**EE (FPC)**	**EE (FP)**
$\text{corr}(sc_f/scid_f, p_f)$	-0.36	-0.30	-0.38	-0.46	-0.36
$\text{corr}(sc_{fp}/scid_{fp}, p_{fp})$	-0.38	-0.27	-0.31	-0.47	-0.40
$\text{corr}(sc_{fpc}/scid_{fpc}, p_{fpc})$	-0.31	-0.14	NA	-0.44	NA
$\text{corr}(sc_f/scid_f, l)$	-0.04	0.03	0.04	-0.05	-0.11
$\text{corr}(sc_{fp}/scid_{fp}, l)$	-0.05	-0.05	0.09	0.01	-0.10
$\text{corr}(sc_{fpc}/scid_{fpc}, l)$	-0.16	-0.25	NA	-0.03	NA
$\text{corr}(p_f, l)$	-0.46	-0.25	-0.37	-0.24	-0.26
$\text{corr}(p_{fp}, l)$	-0.40	-0.25	-0.38	-0.25	-0.26
$\text{corr}(p_{fpc}, l)$	-0.39	-0.26	NA	-0.24	NA
$\frac{\lvert\{alignments: p_{fp} < p_f\}\rvert}{\lvert\{alignments: p_f < p_{fp}\}\rvert}$	1.66	1.91	1.36	2.10	0.88
$\frac{\lvert\{alignments: p_{fpc} < p_f\}\rvert}{\lvert\{alignments: p_f < p_{fpc}\}\rvert}$	1.55	1.79	NA	2.34	NA
$\frac{\lvert\{alignments: p_{fpc} < p_{fp}\}\rvert}{\lvert\{alignments: p_{fp} < p_{fpc}\}\rvert}$	0.86	1.25	NA	1.23	NA

the addition of C seems to produce fewer alignments with low p-values. One reason may be that C is the least detailed of the three sub-ontologies.

4.3.5 Semantic- vs sequence similarity

A question that could be posed is; how does GO-based semantic similarity compare with sequence similarity in terms of performance? Why use semantic similarity in the first place and not sequence similarity? Many of the related approaches to pathway comparison use sequence similarity, so the measure has been proved to be useful. One advantage of our approach using semantic similarity is that it provides a meta-alignment as a biological explanation to the actual alignment consisting of gene products. A sequence similarity based alignment can not alone provide such information. Additionally, the concept of using the three sub-ontologies of GO makes it possible to align with respect to any combination of the three traits molecular function, biological process and cellular component.

In order to compare GO-based semantic similarity with sequence similarity we here perform a benchmark experiment using the orthologous cell cycle pathways in KEGG

for *H. sapiens* and *M. musculus*. This pathway was chosen because it is well studied and fairly large, providing a large set of data for quantitative analyses. Another reason being that the cell cycle pathway was used earlier in this thesis e.g. to assess the performance of the GOTEM method. The purpose of the experiment is to assess how well GOSAP, using its p-value, can separate known path alignments in a pathway graph from unknown path alignments in the same pathway graph. A diagnostic test involving the statistical measures of sensitivity and specificity, is used. The experiment investigates both GO-based semantic similarity and sequence similarity at the amino acid level. An orthologous pathway graph for two species was chosen, since every path in the graph is a known alignment between the two species, making it easy to assess the results. In the current implementation of GOSAP only one optimal alignment is returned when two paths are aligned with each other. There may be several optimal alignments with the same score, and these are all needed in order to perform a fair quantitative analysis especially when comparing algorithms. A second matter is that the option of using gaps in the alignment increases the computational complexity too much, yielding too many possible alignments. Due to these two concerns, all possible paths (i.e. alignments) were extracted from the cell cycle pathway graph and no gaps were used. Hence, the Smith-Waterman algorithm is not used. Instead it is "emulated" by comparing paths of all possible lengths, because all possible ungapped alignments need to be considered in the quantitative analyses.

The orthologous cell cycle graph used contains 55 gene products, 100 edges and 819 paths with a minimum of two and a maximum of nine gene products. Figure 4.8 shows the number of paths as a function of path length. All *M. musculus* paths were compared with all *H. sapiens* paths with the same length, i.e. all length two paths for mouse are compared with all length two paths for human. A total score S is calculated for each comparison according to:

$$S = \sum_{i=1}^{L} s(Q_i, M_i) \tag{4.5}$$

where L is the alignment length, and $s(Q_i, M_i)$ is score when comparing the gene products at position i for the query path Q and the model path M. The score function s can be represented by the semantic similarity scores s_f, s_{fp} and s_{fpc}, which

were described earlier in equations 4.1, 4.3 and 4.4. It would also be possible to generalise this to a weighted score according to equation 4.6, although this was not done in this experiment.

$$S = \sum_{i=1}^{L} w_f \cdot s_f(Q_i, M_i) + w_p \cdot s_p(Q_i, M_i) + w_c \cdot s_c(Q_i, M_i) \tag{4.6}$$

where w_f, w_p and w_c are adjustable weights with the restriction that $\sum w_f + w_p + w_c = 1$. The scoring function s in equation 4.5 can also be represented by similarity between the amino acid sequences of the gene products Q_i and M_i. We used the BLAST (Altschul et al. 1990) based standalone version of BL2SEQ (Tatusova and Madden 1999) with the BLASTP option to calculate the sequence similarity between two protein sequences. One reason for choosing BLAST is that it was also used by Lord et al. (2003a) in their experiments where semantic similarity was correlated with sequence similarity. The default BL2SEQ parameters were used, except that the expectation value threshold was increased from 10 to 10^9, since a score is desired for every gene product pair comparison even if the comparison is insignificant. The bit score of the highest ranking local alignment returned by BL2SEQ was used. The two-logarithm of the bit score was used in order to yield a score interval more similar to the score interval of semantic similarity (equation 4.2). The identity score S_{id} is, like earlier explained, the score obtained when comparing the query path with itself using equation 4.5. The alignment score can be divided with the identity score, yielding a value $\in [0, 1]$, which is used in some of the later analyses.

The total number of path comparisons is 105251, the sum of the number of paths squared in figure 4.8 (91^2+155^2+124^2+184^2+113^2+88^2+56^2+8^2). An alignment score is calculated for each path comparison, and a p-value is calculated using the earlier described method and the Maslov and Sneppen graph randomisation procedure. 100 randomised versions of the *H. sapiens* cell cycle pathway was used. Like for the GOTEM method, we also use ROC analysis where the true positive rate (equation 3.1) is plotted as a function of the false positive rate (equation 3.2). We use the following definitions for true positives (TP), false negatives (FN), false positives (FP), and true negatives (TN):

- TP= Significant alignments ($p \leq p_t$) from A which are in O.

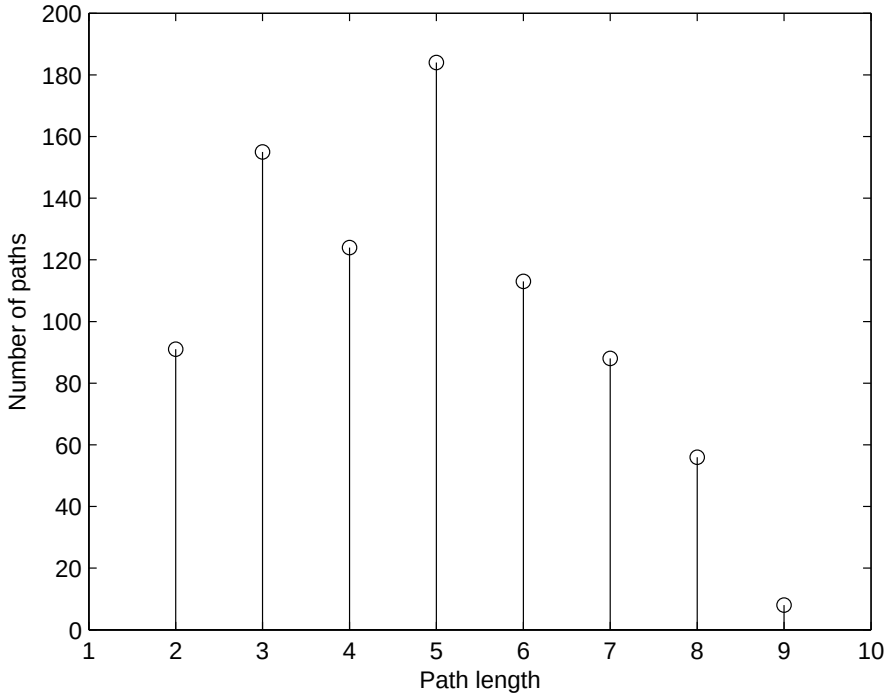


Figure 4.8: Number of paths as a function of path length for the set of all possible orthologous cell cycle paths in *H. sapiens* and *M. musculus*.

- FN= Insignificant alignments ($p > p_t$) from A which are in O.
- FP= Significant alignments ($p \leq p_t$) from A which are not in O.
- TN= Insignificant alignments ($p > p_t$) from A which are not in O.

where p_t is the p-value threshold, A is the set of alignments resulting from the 105251 path comparisons, and O is the set of 819 possible orthologous path alignments in the cell cycle pathway for *H. sapiens* and *M. musculus*. According to this definition, a true positive is a known alignment and a false positive is an unknown alignment. An unknown alignment may represent a new discovery, but biological experiments are needed to validate this.

By varying the p-value threshold in the interval [0,1] in steps of 0.01, the ROC curve in figure 4.9, with an ROC area of 0.924, is obtained when using semantic similarity based on all three GO sub-ontologies to calculate alignment scores. If the cellular component is not used, the ROC area is reduced to 0.913. Only using the

molecular function sub-ontology yields an area of 0.900. This shows that several ontologies combined can better separate known path alignments from unknown path alignments. Using amino acid sequence similarity results in an ROC curve similar to the one for semantic similarity, but with an area of 0.960. These results indicate that GOSAP performs well with respect to sensitivity and specificity using semantic similarity based on all three sub-ontologies, and even better using sequence similarity. Table 4.5 shows the "raw" data that the ROC curves are based on, using the p-value interval [0,0.1]. The table also shows the sensitivity (same as true positive rate) and specificity (the complement of false positive rate). It can be observed that semantic similarity yields higher specificity than sequence similarity for all p-value thresholds. It is also evident that the sensitivity is higher for sequence similarity. For example, for $p_t = 0.02$ there are 213 false negatives for semantic similarity, but only 76 for sequence similarity, explaining the higher sensitivity for sequence similarity. For the same threshold there are 10377 false positives using semantic similarity and 13736 using sequence similarity, which explains the higher specificity for semantic similarity.

In addition to pure ROC analysis, it is of interest to study how the sets of true positives and false positives are related for semantic- and sequence similarity. By studying the results at "set level" it will be possible to see if the measures produce sets which are completely overlapping or complementary. It was found that the true positives captured by semantic similarity are also captured by sequence similarity, i.e. the true positives of semantic similarity are completely overlapping those derived using sequence similarity. However, this is not the case when studying the sets of false positives. The number of alignments in the intersection and differences between the false positive sets for semantic- and sequence similarity is shown in table 4.5. As an example, for the $p_t = 0.02$ threshold there are 7888 alignments in the intersection between semantic- and sequence similarity. There are 2489 alignments captured using semantic similarity, which were not captured using sequence similarity. Conversely, there are 5848 alignments detected by sequence similarity that were not detected by semantic similarity. This shows that the sets of false positives for the two similarity measures are complementary, and that it would be beneficial to utilise both measures

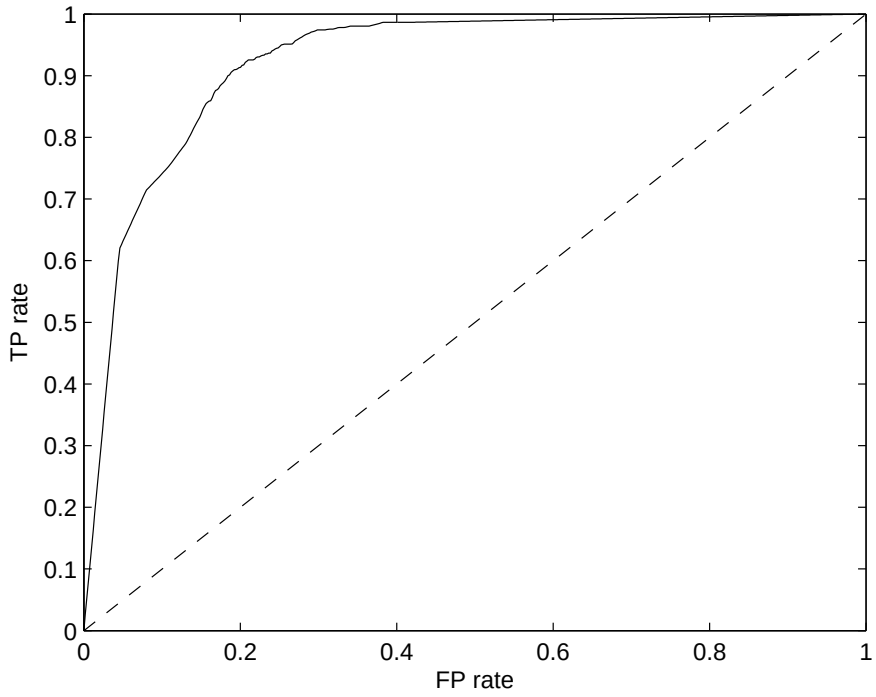


Figure 4.9: ROC curve obtained when using semantic similarity based on all three GO sub-ontologies.

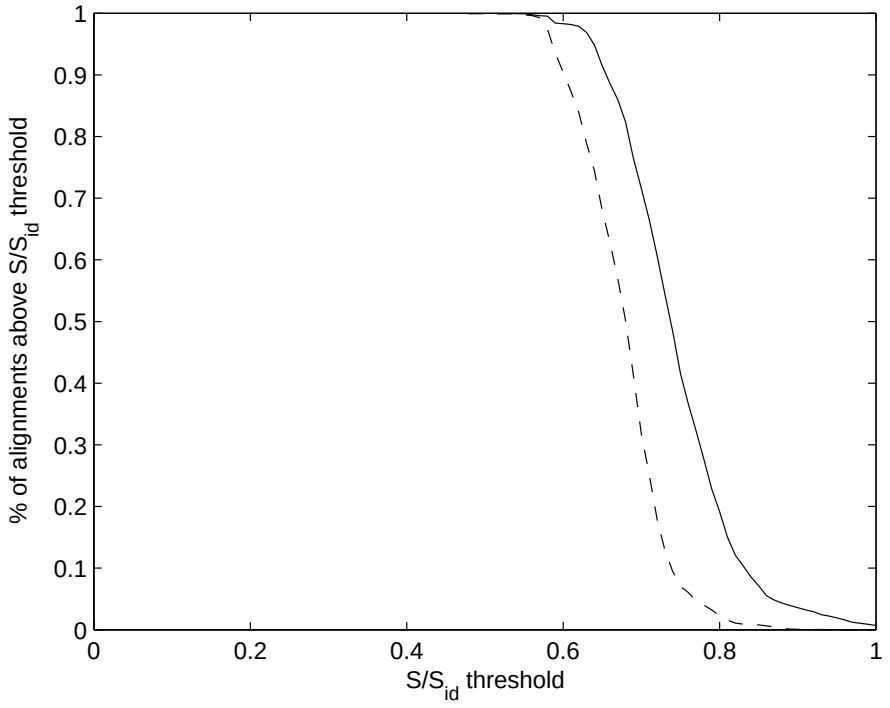


Figure 4.10: Percentage of alignments having $\frac{S}{S_{id}}$ above a certain threshold. Solid line represents the alignments obtained when semantic similarity is used and dashed line shows the corresponding sequence based scores for the same alignments.

in order to capture alignments representing potential new biological discoveries. Even if there are thousands of alignments classified as false positives, it is possible to reduce the number of interesting ones further by ranking them with respect to $\frac{S}{S_{id}}$ for a certain p_t. Figure 4.10 illustrates the percentage of the 2489 "false positive" alignments detected by semantic- but not by sequence similarity, as a function of $\frac{S}{S_{id}}$ threshold for $p_t = 0.02$. The figure also shows the corresponding sequence based scores for the same alignments. It can be observed that approximately 70% of the alignments have $\frac{S}{S_{id}} \geq 0.7$, and that 30% of the corresponding sequence scores are above that threshold. 20% alignments have $\frac{S}{S_{id}} \geq 0.8$ using semantic similarity and 3% of the corresponding sequence based scores.

Figure 4.11 shows the percentage of the 5848 alignments detected using sequence similarity but not semantic similarity, together with the percentage of semantic similarity scores for the same alignments. It can be noted that approximately 80% of

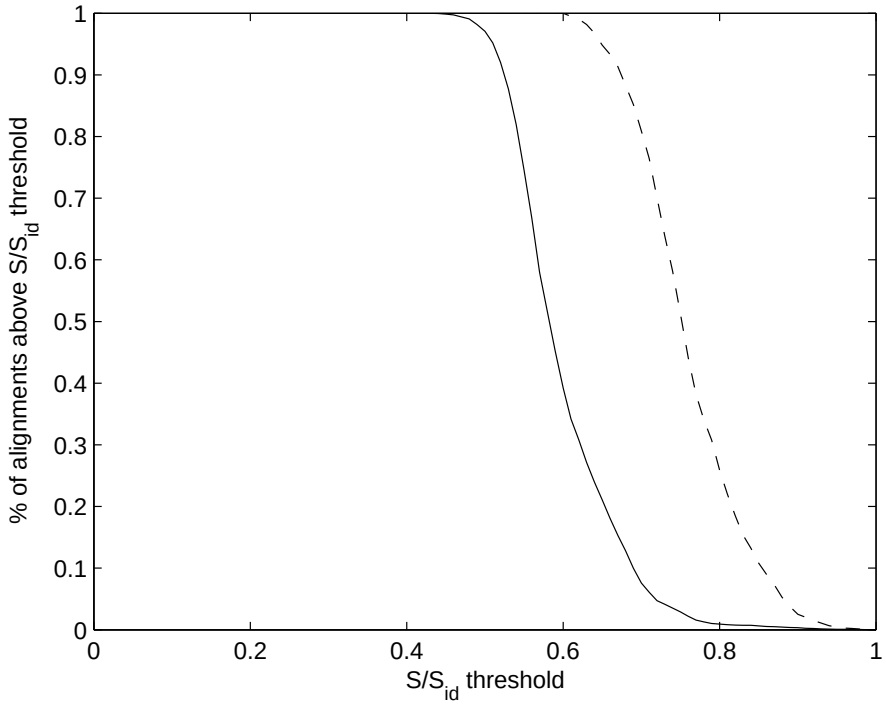


Figure 4.11: Percentage of alignments having $\frac{S}{S_{id}}$ above a certain threshold. Dashed line represents the alignments obtained when sequence similarity is used and solid line shows the corresponding semantic similarity scores for the same alignments.

the alignments have $\frac{S}{S_{id}} \geq 0.7$, and that 8% of the corresponding semantic similarity scores are above that threshold. 26% alignments have $\frac{S}{S_{id}} \geq 0.8$ using sequence similarity and 1% of the corresponding semantic similarity scores. 65 alignments out of 5848 have have $\frac{S}{S_{id}} \geq 0.95$ using sequence similarity.

Table 4.5: Data used to derive the ROC curves. In columns two through seven, each table cell contains values for semantic similarity based on all three GO sub-ontologies (upper value) and sequence similarity (lower value).

p_t	$\|TP\|$	$\|FN\|$	$\|FP\|$	$\|TN\|$	$SENS$	$SPEC$	$\|FP_G \cap FP_S\|$	$\|FP_G - FP_S\|$	$\|FP_S - FP_G\|$
0	507	312	4757	99675	0.6190	0.9544	3898	859	6247
	691	128	10145	94287	0.8437	0.9029			
0.01	585	234	8349	96083	0.7143	0.9201	6402	1947	6022
	729	90	12424	92008	0.8901	0.8810			
0.02	606	213	10377	94055	0.7399	0.9006	7888	2489	5848
	743	76	13736	90696	0.9072	0.8685			
0.03	620	199	11616	92816	0.7570	0.8888	8915	2701	5744
	758	61	14659	89773	0.9255	0.8596			
0.04	634	185	12622	91810	0.7741	0.8791	9696	2926	5615
	765	54	15311	89121	0.9341	0.8534			
0.05	646	173	13554	90878	0.7888	0.8702	10482	3072	5432
	778	41	15914	88518	0.9499	0.8476			
0.06	657	162	14177	90255	0.8022	0.8642	11135	3042	5412
	788	31	16547	87885	0.9621	0.8416			
0.07	669	150	14755	89677	0.8168	0.8587	11693	3062	5334
	801	18	17027	87405	0.9780	0.8370			
0.08	677	142	15185	89247	0.8266	0.8546	12122	3063	5283
	806	13	17405	87027	0.9841	0.8333			
0.09	683	136	15557	88875	0.8339	0.8510	12569	2988	5301
	808	11	17870	86562	0.9866	0.8289			
0.1	683	136	15557	88875	0.8339	0.8510	12953	2980	5302
	808	11	17870	86562	0.9866	0.8289			

4.4 Discussion

We have developed GOSAP, a method for extracting and aligning paths from biological pathways containing gene products, where GO annotation is used for semantic comparison of gene product pairs and GO-based semantic similarity for all three sub-ontologies is used to score alignments. The main contribution of our method is that any kind of biological pathway where nodes are gene products can be aligned. In contrast, the previous methods most related to our approach have focused on metabolic enzyme-to-enzyme pathways utilising the EC hierarchy, which limits their application to metabolic pathways. The application examples presented in this chapter demonstrate the generality of GOSAP and the range of application areas. The results also indicate a larger sensitivity than when using EC enzyme annotation, since the information based on GO annotation is more detailed, and reflects a wider range of properties of the compared gene products. Empirical evidence in section 4.3.5 also suggest that both the sensitivity and specificity of the path alignment process can be improved by combining the function- , process- and component ontologies of GO.

In the application where putative pathways are assessed, GOSAP is potentially useful for correction of query pathway segments that diverge from the model pathway, and also for predicting what gene products may be missing in the query pathway. An intended application area is therefore to use GOSAP alignments as a basis for modifying and correcting putative pathways that have been derived from experimental data using reverse engineering or data mining methods.

Execution times for pathway comparisons on a standard 2 GHz PC varied from a few seconds for a comparison between small metabolic networks involving only a handful of gene products and super-paths, to one hour when comparing the MAPK pathway with the cell cycle pathway (these pathways have 49 and 65 nodes, and 37 and 195 super-paths, respectively). This is when using 100 randomised models in the statistical significance calculation. The speed of the alignment algorithm could possibly be improved by introducing heuristics in order to reduce the computational complexity.

There are different parameter settings in GOSAP that must be considered, espe-

cially the gap penalty and the significance value. Currently, these settings must be set manually by the user. It was for example empirically observed that the significance values for alignments generally increase as a function of decreasing gap penalty, i.e. many alignments involving randomised graphs get higher scores, which in turn makes it harder to get significant alignments. As for the gap penalty, increasing gap penalties promote the matching of less similar gene products in order to avoid gaps, whereas very low penalties (or no penalty at all) result in fragmented alignments where only small sub-segments match. Different gap penalty strategies, such as affine gap costs, would also be of interest to investigate.

The performance of GOSAP depends on the quality of the GO annotations, both regarding experimental evidence (e.g. "traceable author statement") and regarding the specificity of GO terms for individual gene products. Currently, annotations of all evidence types are used, since we found it unfair to disqualify any specific type. But it is generally regarded that "traceable author statement" is the most reliable type of annotation. As for specificity, some gene products are annotated with very specific terms and some are not. GOSAP would benefit from more fine-grained future versions of GO, which would further increase the sensitivity of the approach.

Furthermore, a multiple alignment extension of GOSAP would enable the study of more than two species or paths at a time. It would also be possible to develop a measure of how semantically similar entire pathways are, based on the current method of comparing individual paths in pathways.

4.4.1 Generalisation of GOSAP

One obvious question is how GOSAP can be generalised in order to be of interest to a wider research community. In this section we will discuss this matter with respect to three application domains; 1) the biological pathway domain, which is covered in this thesis, 2) the biological domain, and 3) the non-biological domain.

Biological pathway domain

We have shown that GOSAP in its current form is able to semantically compare paths in molecular pathways using GO, and we have also shown that the addition of amino

acid sequence similarity using the BLAST-based BL2SEQ program would enhance the performance of the algorithm and that the measures complement each other in a synergistic manner. It was found that the measures can be used separately and that the results are complementary. Another approach would be to integrate the measures into a composite measure. However, this has not been investigated.

It would also be possible to add more measures of similarity between gene products. One example is structure similarity. Even though the amino acid sequence is believed to determine the fold of gene products, there are cases when gene products are similar in structure but not in sequence (remote homologs). Hence, structure would add something that the sequence does not reflect. If the 3D structures are known for the two gene products under comparison, there are methods for aligning these and obtaining a measure of structural distance, which can be recalculated as similarity. There are many methods available for protein structure alignment. Some examples are ProFit (McLachlan 1982), DALI (Distance ALIgnment matrix method) proposed by Holm and Sander (1996), and MAMMOTH (MAtching Molecular Models Obtained from THeory) by Ortiz et al. (2002). A common measure of distance between structures is RMSD (Root Mean Square Distance), which is defined as:

$$RMSD_{ab} = \sqrt{\frac{\sum_{i=1}^{N} |p_i^a - p_i^b|^2}{N}} \tag{4.7}$$

where p_i is the spatial position for the atom with index i in the protein structures a and b which use the same number N of atoms for the comparison. Often only the backbone (carbon-α atom chain) is studied, but it is also possible to include all atoms.

It would also be possible to extend GOSAP to support biological pathways with molecules other then gene products, since the ChEBI (Chemical Entities of Biological Interest) ontology covers four different sub-ontologies of small molecules such as metabolites in metabolic pathways, and also other kinds of chemical entities such as subatomic particles (Degtyarenko et al. 2008). The sub-ontologies are molecular structure, biological role, application, and subatomic particle. Molecular structure classifies molecules according to entity, or group. Example of an entity abstraction hierarchy is “inorganic radical anions [is-a] inorganic radical ions [is-a] radical ions [is-a] ions [is-a] molecular entities”. An example group hierarchy is “divalent carboacyl

groups [is-a] carboacyl groups [is-a] acyl groups [is-a] organic groups [is-a] groups". The biological role sub-ontology defines the role of the molecule for a certain biological context. An example hierarchy is "lupinic acid [is-a] purine alkaloids [is-a] alkaloids [is-a] secondary metabolite [is-a] metabolite [is-a] biological role". Application is the sub-ontology covering the intended way humans would use a molecule. An example of this is "macrolide insecticide [is-a] antibiotic insecticide [is-a] insecticide [is-a] pesticide [is-a] application". The subatomic particle sub-ontology classifies particles that are smaller than atoms. The same kind of semantic similarity calculations as in GOSAP could be used in combination with ChEBI. A large number of chemical entities are annotated with the ChEBI terms, so the GOSAP procedure for calculating term probabilities could be employed.

Biological domain

GOSAP could potentially be applied to food chains and food webs, a research field in theoretical ecology. A food chain is a path describing how different species within an ecosystem are related with respect to feeding, i.e. what species consumes what species. This can also be described as an energy flow between species (Morin 1999). An example food chain is $Grass \rightarrow Cricket \rightarrow Frog \rightarrow Snake \rightarrow Bird \rightarrow Fungi \rightarrow Grass$, where an arrow means "is consumed by". Of course, there are different kinds (species) of grass, crickets, frogs, snakes, birds and fungi. A food web is a graph where one species can be consumed by several other species, and each species can also consume several species. By using the path extraction methodology in GOSAP, it would be possible to convert a food web to a set of food chains, which would correspond to the set of super-paths in the food web.

It is possible to classify biological species using classic taxonomy hierarchies like the Linnean taxonomy hierarchy (Margulis and Schwartz 1997), which has been modified over the years and nowadays contains eight hierarchically organised layers. The layers are $Domain \rightarrow Kingdom \rightarrow Phylum \rightarrow Class \rightarrow Order \rightarrow Family \rightarrow Genus \rightarrow Species$, where the arrow shows order from root to leaf. Possible domains are Archaea, Eubacteria and Eukaryota. Furthermore, there are currently six kingdoms to classify species; Archaea, Monera, Protoctista, Fungi, Plantae and Animalia. Layers beneath

kingdom provide an increasingly more fine grained categorisation until the leaf node is reached, which represents a species. A species would be analogous to a gene product in the biological pathway domain, and groupings for each taxonomic layer would correspond to GO terms. An annotation of a gene product to a GO term would correspond to a species associated with a specific genus. The set of all classified species could be used in the same kind of term probability calculation as for GOSAP, and semantic similarity measures could be applied in the same way. By applying the GOSAP path alignment method to food chains, it would be possible to find similarities between food chains in different ecosystems and biotopes. For example, there may be interesting similarities between food chains occuring in the Pacific ocean and in the Mediterranean, or in chains found in rain forests in different parts of the world. The concept of meta-alignment, inherent to GOSAP, gives information on what different food chains have in common. Some positions of a food chain alignment may have a certain genus as the minimum subsumer, and other positions may contain for example a specific family or order. Like for the biological pathway domain, it would be possible to complement semantic similarity with other kinds of species similarity methods like phylogeny by sequence similarity, morphology and functional characteristics.

For the biological domain there is an increasing number of biological ontologies maintained by the OBO (Open Biomedical Ontologies) foundry which share a common set of principles for ontology design, yielding a set of interoperable ontologies (Smith et al. 2007). GO is part of OBO, and other examples of mature ontologies are the cell ontology (CL), covering different cells from prokaryotic to mammalian, and the foundational model of anatomy (FMA), which describes the body structure for mammals and in particular humans. There are also less mature ontologies, that are still under development. The ontologies of the OBO foundry could be useful in other, yet unidentified, applications of GOSAP.

Non-biological domain

We also propose that the GOSAP methodology for aligning paths can be useful in the study of social networks. A social network is a graph where the nodes are social entities such as human individuals, groups of individuals, corporations and coun-

tries (Wasserman and Faust 1994). Edges represent different types of relationships between the social entities, e.g. communication, transactions and trade. Social network analysis is about identifying regularities and patterns among the entities that are interacting. The research field of social network analysis is going back about 50 years and is very diversified. Usually the analysis consists of the application of graph topology based measures such as degree centrality, betweeness centrality and closeness centrality. These measures are helpful in order to identify e.g. which nodes are most connected to other nodes, which nodes are important mediators to other parts of the graph, and what nodes that can access all other nodes with as short paths as possible. It is easy to realise that such measures are useful in e.g. terrorist networks created using intelligence or public information in order to analyse the organisational structure and identify important key entities. The concept of semantic path alignment could be a contribution to this field in order to find similar interaction patterns between paths of social entities representing e.g. different organisations or collaboration networks. Ontologies or abstraction hierarchies covering the traits of social entities could be used for semantic calculations. For the domain of warfare and terrorism there are several taxonomies available, e.g. the ones provided by Cycorp Inc. (taxonomies.cyc.com), with thousands of concepts such as organisations, weapons, terrorists, terrorist groups and terrorist attacks. Using the GOSAP methodology, newly discovered putative terrorist networks could be semantically aligned with other known terrorist networks e.g. in order to discover dangerous patterns of activity.

Chapter 5

GOSAM: Gene Ontology based Semantic Alignment of biological pathways by gene product Mapping

A large number of biological pathways have been elucidated recently. Examples are the metabolic and regulatory pathway collections in KEGG (Kanehisa and Goto 2000), the metabolic pathways for *E. coli* in EcoCyc (Karp et al. 2004) and in MetaCyc (Caspi et al. 2008) for many other organisms. There is a need for methods to analyse these pathways. One class of methods compares pathways semantically, in order to discover parts that are evolutionarily conserved between species or to discover intra-species similarites. Such methods usually require that the topologies of the pathways being compared are known, i.e. that a query pathway is being aligned to a model pathway. However, sometimes the query only consists of an unordered set of gene products. Previous methods for mapping sets of gene products onto known pathways have not been based on semantic comparison of gene products using ontologies or other abstraction hierarchies.

Therefore, we here propose an approach that uses a similarity function defined over Gene Ontology (GO) terms to find semantic alignments when comparing paths in biological pathways where the nodes are gene products. A known pathway graph is used as a model, and a search algorithm is used to find putative paths using a set of experimentally determined

gene products. The method uses a measure of GO term similarity to calculate a match score between gene products, and the fitness value of each candidate path alignment is derived from these match scores. A statistical test is used to assess the significance of the derived alignments. The performance of the method has been tested using different kinds of search algorithms, and regulatory pathways for *S. cerevisiae* and *M. musculus*.

5.1 Introduction

The number of biological pathways that have been experimentally elucidated or computationally predicted is growing rapidly. Hence, there is a great need for methods to compare pathways, so that similarities and differences can be analysed both within and between different species. Just as sequence alignments may help identifying evolutionary changes such as insertions, deletions and substitutions, a pathway alignment may help identifying evolutionary events at the pathway level, such as gene duplication and divergence of function. An alignment of two similar pathways from the same species may for example suggest that the aligned pathways have evolved from a common ancestor pathway by gene duplication followed by divergence (Pinter et al. 2005).

Of particular interest is the class of methods that compare pathways semantically, i.e. using the annotation of the pathway components to discover homologies that are based on similarities regarding functional role, biological process or cellular location. Most previous work on such methods has focused on metabolic pathways, utilizing the hierarchy of the enzyme nomenclature[1] to calculate match scores (Dandekar et al. 1999, Tohsato et al. 2000, Pinter et al. 2005, Wernicke and Rasche 2007). It has also been assumed previously that the comparison is done between pathways with known topologies. See previous chapter for more information on such methods. However, sometimes only a set of gene products is available on the query-side, without any knowledge about how the gene products interact, and the goal is to derive a putative pathway by finding the best possible matching of gene products onto the known model pathway. Some earlier methods (Dahlquist et al. 2002, Karp et al. 2002, Doniger et al. 2003, Chung et al. 2004) for mapping gene products onto known pathways do this merely for presentation purposes using gene product associated experimental results (e.g. microarray gene expression data) and these methods are not based on approximate matching using abstraction hierarchies or ontologies. Gene Set Enrichment Analysis (GSEA) is a related method which returns sets of genes from e.g. pathways or complexes that are coordinately expressed in a microarray

[1] www.expasy.org/enzyme

experiment (Mootha et al. 2003, Subramanian et al. 2005). However, the topology of the pathways is not considered, and no semantic matching is performed.

We therefore propose GOSAM, which is a Gene Ontology-based method for finding semantic alignments between paths in biological pathways where the nodes are gene products. GOSAM uses a known pathway graph, from which a set of model paths are extracted. It then uses a search algorithm to derive putative paths, semantically similar to the model paths, from a set of experimentally derived gene products. We compare the performance of different types of search algorithms using benchmark experiments, and derive example alignments in a cross-species experiment. GOSAM can for example be used to semantically map differentially expressed genes identified in a microarray experiment onto known regulatory pathways. This is particularly useful if the experiments have been conducted on a species where little is known about its pathways. It would also be possible to use gene products identified using other experimental methods, or to manually add gene products that are known to be important. GOSAM is published in Gamalielsson and Olsson (2007) and Gamalielsson and Olsson (2008a), but is referred to as EGOSAP in these publications, since the tested search algorithm was an evolutionary algorithm (EA). In this thesis, the possibility of using different search algorithms is emphasised, and the performance of various algorithms is compared.

5.1.1 Related work

There are tools that are capable of mapping groups of gene products onto known pathways, but they do so using only the identity of gene products and for visualisation purposes. An example is GenMAPP (Dahlquist et al. 2002), which is a tool where the genes and their colour-coded expression values are mapped onto known pathways. It is also possible to construct and edit pathway diagrams in GenMAPP. Initially, a set of pathways were provided by the authors, but the idea is that the users of the tool will contribute pathways as well. The gene expression data is imported and fold change with accompanying p-value can be calculated given control and test conditions. Apart from having a database of pathways, GenMAPP also has a gene identifier database which supports the major formats such as Affymetrix gene names and Genbank. In

addition, the genes in the pathway diagrams are linked to other public databases. No particular results from experiments are shown in the paper.

There is also the GenMAPP accessory software MAPPFinder (Doniger et al. 2003) where the Gene Ontology is used in combination with microarray gene expression data in order to derive the amount of genes that are changed for each GO term in all three sub-ontologies. A change criterion is defined by the user, e.g. a twofold change in expression. For each GO term, MAPPFinder derives the percentage of measured genes that are annotated with the term and also meet the change criterion. This is defined as number of genes measured and changed divided by the number of genes measured and annotated with the term. Another statistic is the percentage of genes measured for the term. This is derived by dividing the number of genes measured and annotated with the term with the total number of genes in the genome that are annotated with the GO term. The two described statistics are also applied to the aggregation of all the descendant terms to the GO term under study, referred to as "nested" results in the paper. Additionally, a z-score is calculated for each term, which indicates if the number of genes conforming to the change criterion is significantly different from the number expected by chance. It is defined as the fraction between the difference between the observed and expected number of genes, and the standard deviation of the observed number of genes. The utility of the tool was demonstrated by applying it to a *M. musculus* dataset of cardiac development, maturation and aging. The results are not discussed here since they are not interesting in the context of this thesis.

ArrayXPath (Chung et al. 2004) is a similar tool where gene expression clusters can be mapped onto the best matching pathways in a database. The tool collects pathways used for comparison from major databases such as KEGG, GenMAPP (also a tool) and Biocarta. Input data is a text file where each row contains a gene, its cluster identity and optionally its gene expression values at different time points. The tool does not perform any clustering, but expression values can be used in the presentation. Initially, Statistics are derived showing used pathway databases, the number of pathways in each database and the number of nodes. Additional statistics cover number of nodes in the input data that are represented in the used pathways. Subsequently, each pathway in the pathway database of ArrayXPath is searched for

the presence of the genes in each of the clusters in the input file. If there is a pathway hit, the number of matched nodes in the pathway is presented together with the total number of non-redundant and redundant nodes in the pathway. A measure of statistical significance, the q-value (Storey and Tibshirani 2003), is used to assess each pathway hit. The q-value is an extension to the false discovery rate. Each pathway in the "hit-list" can be graphically inspected where the matched input nodes of the clusters are highlighted.

Gene Set Enrichment Analysis (GSEA) is method which can detect coordinately expressed sets of genes in e.g. pathways or earlier established groups of genes (Mootha et al. 2003, Subramanian et al. 2005). When a microarray experiment has been performed, the genes can be sorted into a list L according to differential expression between two conditions. Studying expression changes for single genes may not be useful in the context of pathway analysis, since it is often more important to detect that a set of genes in a pathway are changed a little than a single gene in the same pathway is changed a lot. A coordinated change of a gene ensemble has usually a bigger impact on the behaviour of a cellular process. Additionally, it is possible that modest expression changes for single genes are hard to detect due to the inherent noise in the microarray technology. GSEA determines if the genes in a gene set S are randomly distributed in the sorted list L or if they are located at the top or bottom of the list. In the latter case, the genes in S and L are assumed to be related. There are three major steps in GSEA. The first step involves the calculation of the enrichment score ES, which indicates the degree to which S is overrepresented at the top or bottom of L. This is done by increasing a running sum with a contribution c_p if a gene in L is present in S, and if not present the running sum is decreased with c_n (Mootha et al. 2003). c_p is defined as $c_p = \sqrt{\frac{|L|-|S|}{|S|}}$, and c_n is calculated as $c_n = \sqrt{\frac{|S|}{|L|-|S|}}$. The maximum of the running sum over all positions in L is the enrichment score ES. Step two in GSEA is to assess the statistical significance of ES. This is done by permuting the class labels of the genes 1000 times, and each time performing an enrichment score calculation again for all gene sets. By counting the number of occurences of scores higher or equal than the original score, a nominal p-value can be calculated. The third step involves an adjustment of the nominal p-value

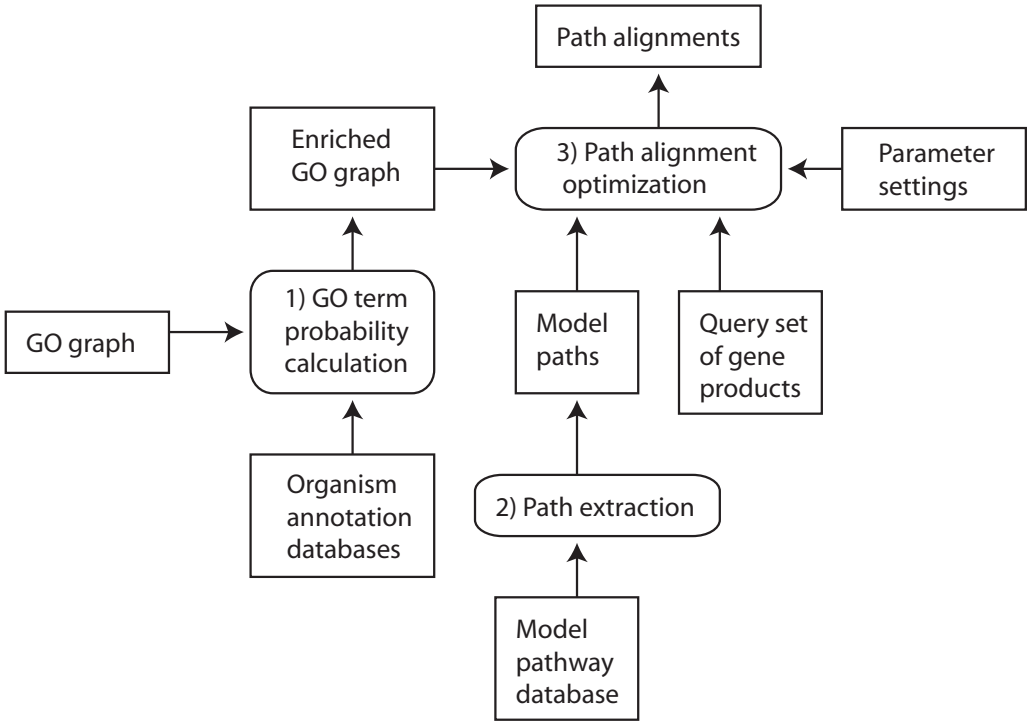


Figure 5.1: The GOSAM method. Boxes with rounded corners represent procedures, and rectangular boxes represent information.

to account for multiple hypothesis testing.

5.2 Method

GOSAM, which is summarised in figure 5.1, is similar to the GOSAP method in the previous chapter which compares a user-specified query pathway graph with a model pathway graph, using three procedures. The first two are preparatory procedures used to set up the GO annotation data and the model paths in such a way that semantic alignments can be derived. These two initial procedures are identical for GOSAP and GOSAM, and are therefore described in less detail here.

1) GO term probability calculation

For every GO annotation term, a probability is calculated using an annotation database for one or several organisms. These probabilities reflect the frequencies with which the annotation terms occur, and are used in the alignment procedure to calculate the semantic similarity of each pair of gene products. This is based on the observation

that more specific terms tend to have lower GO term probabilities, when these are calculated using the procedure proposed by Lord et al. (2003a), which was introduced in section 2.4.1, used in GOSAP in section 4.2.1, and also reproduced here for clarity and convenience:

For each gene product G_i in an annotation database D:

Increment a counter C_j for each GO term T_j appearing in the annotation of G_i, and increment the corresponding counter of each ancestor term of T_j.

For each GO term T_k in GO:

Calculate the term probability $p(T_k) = \frac{C_k}{N}$, where N is total number of annotations in D.

In the example in figure 5.3 (discussed in more detail later), the term probabilities appear as a p value for each GO term.

2) Path extraction

An algorithm involving depth-first search is used to derive all model paths originating from each node in the model pathway graph. Extension of a path ends whenever a leaf node or a previously visited node is encountered (so that cycles are handled). Furthermore, only the *super-paths* are used in the subsequent path alignment, i.e. the set of paths such that no path is included in its entirety as a sub-path of another path. The purpose is to obtain a minimal set of paths, while still covering the entire pathway graph. Putative paths are evolved for each of the extracted paths in the path alignment optimisation procedure.

3) Path alignment optimisation

The original GOSAP method aligns pairs of paths using the Smith-Waterman(Smith and Waterman 1981) algorithm with match scores calculated by a semantic similarity function over the "alphabet" of Gene Ontology annotation terms. However, this relies on the assumption that the topology of both the query- and model pathway graph is known. As mentioned earlier, sometimes only a query set of gene products is available, and there is no knowledge available about how the gene products interact. Therefore, the purpose of GOSAM is to suggest putative paths using the query set of gene products. A search algorithm derives paths that are semantically similar to paths in the

model pathway, and it is assumed that paths are permutations of gene products from the query set, i.e. a gene product can only appear once. This is the case in e.g. gene regulatory networks. We evaluate six different search algorithms for path alignment optimisation; an evolutionary algorithm (EA), simulated annealing (SA), stochastic hill-climber (SHC), iterated hill-climber (HC), greedy search (GS) and random search (RS). We also theoretically estimate the performance of enumeration for solving the problem. The application of each of the search algorithm alternatives are described in the following.

When using an EA, the path alignment optimisation works like illustrated in figure 5.2. As input, GOSAM takes a query set of gene products derived from experimental data (upper left in figure 5.2) and a model pathway graph obtained from a pathway database, e.g. KEGG(Kanehisa and Goto 2000) (upper right). The extracted paths are submitted one at a time to be aligned (indicated by filled circles). In order to search for an optimal alignment, GOSAM samples the query set of gene products to generate an initial random population of candidate paths. Each individual is represented as an initially random permutation of gene products from the query set with the same length as the model path. The maximum length of the candidate paths is the same as the length of the model path, and in order to allow gaps, a special symbol is used to signify "no gene product" (indicated by circles with dashed borders). A user-defined number of "no gene product" symbols can be added to the query set. The match score between a specific model gene product (MGP) and a "no gene product" symbol in the derived path is set to the average semantic similarity between the MGP and all gene products in the query alphabet.

During a number of iterations, the population of candidate paths is replaced by a new population by fitness-based tournament selection of "parent" paths, from which "offspring" paths are generated using recombination and mutation. Binary tournament selection is used, which means that two random individuals are drawn from the population and the best of these is selected(Michalewicz and Fogel 2004). Selection is done without replacement, i.e. each individual can be selected several times for the next generation. Variation operators used are partially mapped crossover (PMX) and a mutation operator, and are performed with probabilities p_c (crossover) and p_m

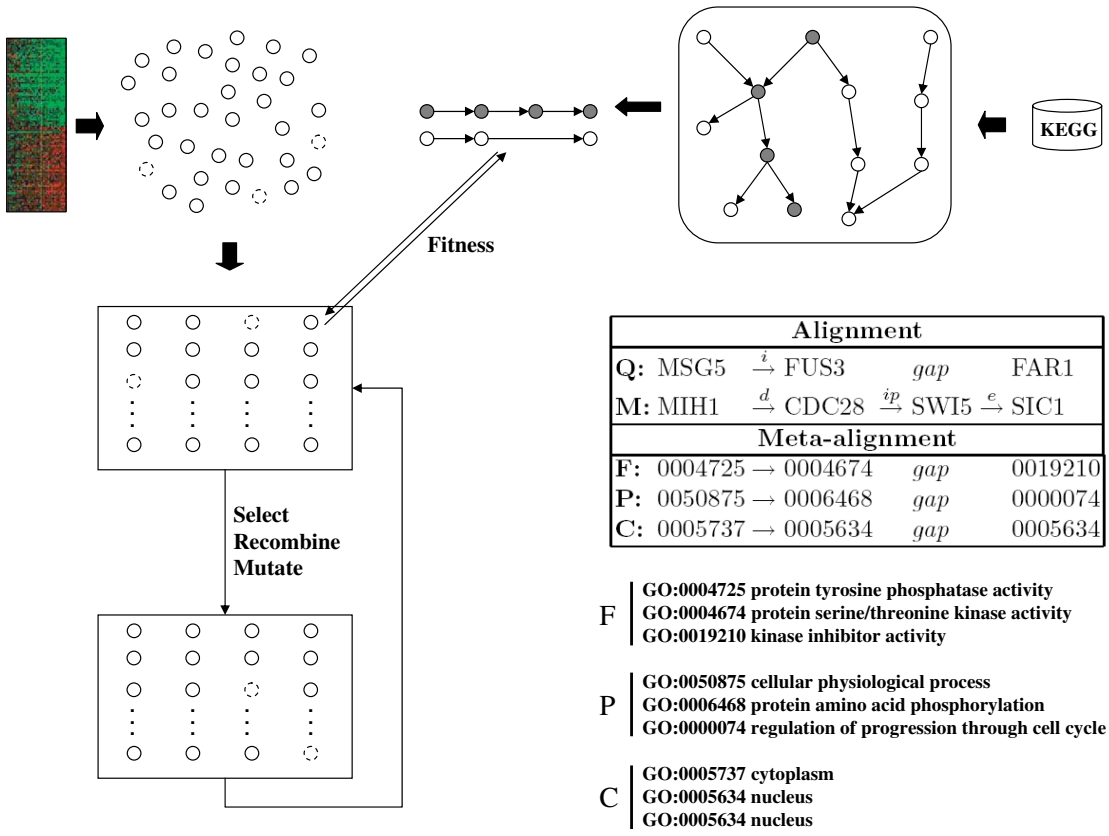


Figure 5.2: Path alignment optimisation procedure in GOSAM using an evolutionary algorithm. For explanations, see text.

(mutation). PMX operates on two parents where a one-to-one mapping between the gene products of the two parents is created at a randomly chosen middle segment (Michalewicz and Fogel 2004). This mapping is used during crossover to ensure that feasible offspring are generated, i.e. an ordered set of elements with no duplicates. Mutation is done by a combined operator, which either (with 50% probability) switches the gene products at two random positions in a parent, or replaces the gene product at a randomly chosen position with a random one from the query set, if possible. An elitist strategy is used where the worst individual of the old population is replaced by the best individual in the current population. This enforces a monotonous fitness growth of the population's best fitness.

The fitness of each alignment is calculated from the semantic similarities of the aligned pairs of gene products, according to the following equation:

$$F = \frac{\sum_{i=1}^{L} w_f \cdot s_f(M_i, Q_i) + w_p \cdot s_p(M_i, Q_i) + w_c \cdot s_c(M_i, Q_i)}{\sum_{i=1}^{L} w_f \cdot s_f(M_i, M_i) + w_p \cdot s_p(M_i, M_i) + w_c \cdot s_c(M_i, M_i)} \tag{5.1}$$

where L is the alignment length, and the weights w_f, w_p and w_c are adjustable with the restriction that $\sum w_f + w_p + w_c = 1$. s_f is the molecular function semantic similarity between gene products at position i for the model path M and query path Q. s_p and s_c are the respective measures for the biological process and cellular component ontologies. The denominator part of equation 5.1 enforces the fitness interval [0,1]. s_f is calculated according to equation 5.2, which is similar to the one defined by Lord et al. (2003a). Also s_p and s_c are calculated according to equation 5.2, but using the biological process and cellular component annotations, respectively.

$$s_f(M_i, Q_i) = max(\{SS(T_k, T_l) : T_k \in t(M_i), T_l \in t(Q_i)\}) \tag{5.2}$$

$$SS(T_k, T_l) = -log_2(p_{ms}(T_k, T_l)) \tag{5.3}$$

In equation 5.2, $t(M_i)$ and $t(Q_i)$ are the sets of GO annotations for M_i and Q_i. The fitness function promotes individuals which are semantically similar to the model sequence with respect to all three sub-ontologies. In equation 5.3 (defined by Resnik (1999), earlier used in this thesis in equations 2.1 and 4.2, but also reproduced here as equation 5.3 for clarity and convenience), $p_{ms}(T_k, T_l)$ is the probability of the minimum subsumer for GO terms T_k and T_l. The minimum subsumer ms is the ancestor term with lowest probability that terms T_k and T_l have in common.

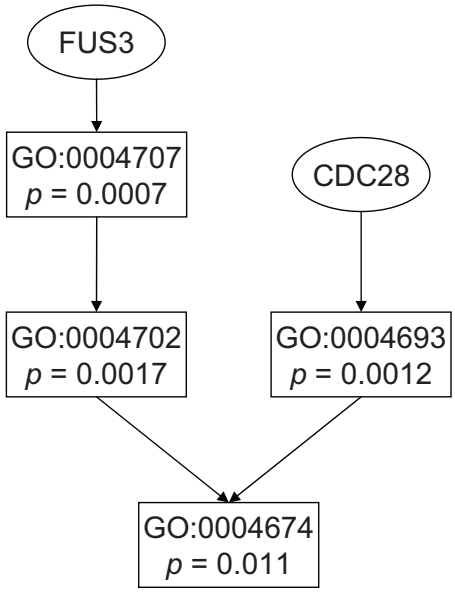


Figure 5.3: Gene products (ovals) mapped to GO terms (rectangles) according to their molecular function annotation. Higher located GO terms are more specific (lower p) than lower located terms. GO:0004674 is the minimum subsumer of GO:0004707 and GO:0004693.

The semantic similarity calculation for two example gene products, FUS3 and CDC28, is illustrated in figure 5.3. FUS3 is annotated with GO term GO:0004707 and CDC28 is annotated with GO:0004693. The minimum subsumer of these two terms is GO:0004674, and the probability of this GO term (reflecting the frequency with which it is used in the annotation database) is 0.011. Hence, the semantic similarity between FUS3 and CDC28 is $-log_2(0.011) \approx 6.5$.

In the box to the middle right in figure 5.2 is shown the alignment between a query path, Q, and a model path, M. Below the alignment is shown a meta-alignment, indicating which GO terms have been used in the calculation of fitness, and below the box is shown the meaning of the annotatation terms from the three GO sub-ontologies for molecular function (F), biological process (P) and cellular component (C).

The other algorithms for performing the path alignment optimisation are described in detail in section 2.6.2. The main difference from the EA is that the non-evolutionary algorithms only maintain and optimise one solution at a time. Like for the EA, all algorithms except greedy search initially create a random permutation of gene products as a starting point. Another similarity is that all algorithms except the

greedy algorithm and random search use the same mutation operator as the EA in order to explore the search space. The greedy algorithm systematically matches the best gene product in the query alphabet to each gene product in the model sequence. During random search, the algorithm just jumps from one point in the search space to another by generating a completely new solution. Furthermore, the same fitness function as for the EA, was used for all search algorithms except greedy search. As the greedy search algorithm does not operate on complete solutions, the score is accumulated during the search until a complete solution has been derived. Apart from this matter, the scoring is identical to the scoring of the other algorithms.

Statistical significance of alignments

The alignment score itself may not be sufficient for judging the quality of an alignment. Therefore, an assessment of the statistical significance of alignments was performed according to the procedure described by Maslov and Sneppen (2002). In this procedure two edges $A \rightarrow B$ and $C \rightarrow D$ are randomly selected in a graph and rewired into $A \rightarrow D$ and $C \rightarrow B$. If the resulting edges are already present in the graph, a new pair of edges is selected. Hence, a randomisation takes place while preserving the cardinality of each node. A series of random edge switches results in a randomised graph, with the restriction that the randomised graph must be different from the original graph. In GOSAM, a query path can be aligned with a large number of randomised versions of the model pathway using the Maslov and Sneppen procedure. The p-value of an alignment is defined as the fraction of randomised graphs that produce an alignment with equal or higher score than the original alignment. Low p-values are therefore desirable.

5.3 Results

5.3.1 On the complexity

GOSAM aims to search for an optimal alignment to a path of length L, by selecting an ordered subset (i.e. a permutation) of gene products from a gene product "alphabet" of size N. For a permutation, i.e. a candidate solution, the fitness is calculated

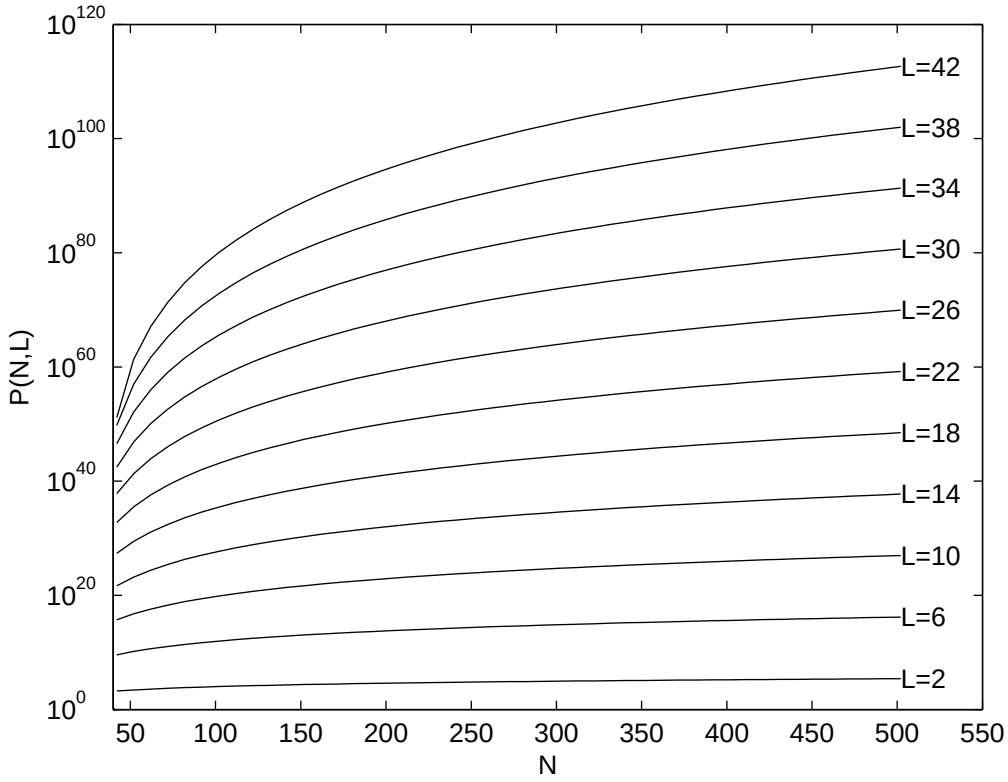


Figure 5.4: Number of possible permutations for different sizes of alphabet (N) and path length (L).

according to equation 5.1. The number of possible permutations of length L that can be generated from a query alphabet of size N is given by: (Rosen 1995)

$$P(N, L) = N(N-1)(N-2)...(N-L+1) = \frac{N!}{(N-L)!} \tag{5.4}$$

Figure 5.4 shows the number of possible permutations for different combinations of N and L. It can be observed that $P(N, L)$ increases exponentially relative to L, and that the impact of N increases for larger values of L. Theoretically, one could enumerate all possible solutions in order to find the optimal one. However, the limitations of computers make exhaustive enumeration of all possible candidate solutions infeasible for paths of biologically realistic lengths.

The complexity of the evaluation function is $O(L)$, where L is the length of the permutation of gene products being optimised. This assumes that the similarity function between two gene products has been implemented as a hash matrix with

constant complexity $O(1)$ for lookup. This evalutation function is used by all search algorithms, except for greedy search where the scoring is more tightly integrated into the algorithm. The complexities of the used search algorithms are specified in the following and should be multiplied with the evaluation function complexity to obtain the total complexity. It should also be kept in mind that it is not possible to assess search algorithms simply by studying the complexity. The performance depends on e.g. the search space, representation of solutions and choice of algorithm parameters. That is a strong reason for actually comparing the search algorithms empirically in our study. Random search (fig 2.7) has a complexity of $O(N)$, where N is number of iterations. Even if $O(N)$ sounds good, the search strategy is very simple and usually returns poor solutions in terms of quality. The complexity of greedy search (see figure 2.8) is $O(N_r \cdot |P_M| \cdot |Q|)$, where P_M is the model path of gene products, and Q is the set of query gene products, and N_r is number of restarts of the algorithm with different permutations of the query set. The iterated hill-climbing algorithm (fig 2.9) used in our study has the complexity $O(N \cdot M)$, where N is number of restarts and M is number of neighbourhood tries for each iteration. The complexity of the stochastic hill-climber (fig 2.10) is $O(N)$, where N is number of iterations. Our simulated annealing algorithm (see figure 2.11) has a complexity of $O(\frac{log(\frac{T_{min}}{T})}{log(r)} \cdot N)$, where T_{min} is the final temperature, T is the start temperature, r is the temperature decrease ratio, and N is number of inner iterations in the algorithm. Since the minimum temperature was defined as $T_{min} = T \cdot r^i$ in section 2.6.2, the number of temperature iterations i can be deduced as described in the following:

$$
\begin{aligned}
T_{min} &= T \cdot r^i \Rightarrow \\
r^i &= \frac{T_{min}}{T} \Rightarrow \\
log(r^i) &= log(\frac{T_{min}}{T}) \Rightarrow \\
i \cdot log(r) &= log(\frac{T_{min}}{T}) \Rightarrow \\
i &= \frac{log(\frac{T_{min}}{T})}{log(r)}
\end{aligned}
$$

The EA has a complexity of $O(N_g \cdot N_i)$, where N_g is number of generations and N_i is number of individuals in the population. Additionally, for all search algorithms except greedy search there are one or several variation operators, like the mutation operator and the partially mapped crossover operator. Such operators have linear complexity, proportional to the number of gene products in the path. However, an increase in path length may not necessarily result in increased execution time, since the probability of e.g. mutation is often adjusted so that a certain number of mutated gene products are expected irrespective of path length.

5.3.2 Datasets

We used three different model graphs for *S. cerevisiae*. The first is a graph created using the transcriptional regulatory chain motifs described by Lee et al. (2002) This graph contains experimentally determined interactions between transcriptional elements. In our study we only use gene products which have annotations for all three sub-ontologies of GO. With this restriction the graph contains 64 gene products, 77 edges and 105 super-paths. Path lengths vary from two gene products to five, with an average path length of 3.3. The second model graph is the cell cycle regulatory pathway from KEGG containing 61 gene products, 81 edges and 151 super-paths. Path lengths vary from two to ten gene products, with an average length of 6.7. The third model is the MAPK signalling pathway from the same database containing 48 gene products, 49 edges and 33 super-paths. Here, path lengths vary from three to eight, with an average length of 6.1.

Two query sets were used containing the products of *M. musculus* genes that were found to be differentially expressed in an experiment comparing transgenic and knock-out mice with wild-type mice (for details regarding the experimental protocol, see Nilsson et al. (2006)). The transgenic query set contained 460 gene products derived from 531 microarray probes, but since gene product annotation from all three GO sub-ontologies is desired, the number of gene products was reduced to 211. For the knock-out query set, the number of gene products was reduced from 256 (284 probes) to 119. In some of the cross-species experiments in section 5.3.4, the gene products from the corresponding *M. musculus* cell cycle and MAPK pathways are

used in conjunction with the *M. musculus* transgenic and knockout datasets. The *M. musculus* cell cycle dataset contains 75 gene products, of which 59 have annotation from all three GO sub-ontologies. The corresponding figures for the MAPK dataset are 121 and 79.

It should be mentioned that GOSAM works with only one GO sub-ontology, i.e. all three are not required. In our examples we chose to use all three sub-ontologies in order to be able to show as informative alignments as possible in our results. If only the molecular function sub-ontology was used, fewer gene products would typically be disqualified due to lacking GO annotation. The Lee et al. (2002) graph would contain 67 gene products, the cell cycle pathway 62, the MAPK pathway 49, and the *M. musculus* transgenic and knockout datasets would hold 331 and 192 gene products, respectively. Hence, requiring only one sub-ontology typically leads to inclusion of more genes, and thus potentially more alignments, but also to less reliable results. Conversely, requiring additional sub-ontologies may reduce the quantity of results, while on the other hand increasing their reliability.

We define a measure H of *semantic homogeneity* in order to assess how similar the gene products are to each other in a gene product set GP:

$$H(GP) = \frac{2}{N(N-1)} \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} SS_{GP}(GP_i, GP_j, w) \tag{5.5}$$

$$SS_{GP}(GP_i, GP_j, w) = w_1 \cdot s_f(GP_i, GP_j) + w_2 \cdot s_p(GP_i, GP_j) + w_3 \cdot s_c(GP_i, GP_j) \tag{5.6}$$

where N is size of the gene product set GP, w is a weight vector, and GP_x is a gene product from a gene product set GP. s_f is the semantic molecular function similarity function according to equation 5.2. s_p and s_c are the corresponding functions for biological process and cellular component, respectively. The H measure computes the average semantic similarity between all two-combinations of gene products from GP.

Applying the measure to the set of gene products in the Lee et al. (2002) model graph yields $H(A) = 4.17$ using $w_f = w_p = w_c = \frac{1}{3}$. The corresponding value for the transgenic *M. musculus* dataset is $H(A) = 1.90$ using the same weight set. This implies that the Lee et al. (2002) dataset is more semantically homogeneous than the transgenic *M. musculus* dataset. This is not surprising since the first dataset only

contains transcriptional regulatory gene products, and the second dataset is more diversified. This measure can help to explain the results from the benchmarking tests in section 5.3.3.

5.3.3 Benchmark experiments

In order to assess under what conditions GOSAM is able to derive optimal alignments, a number of benchmark simulations were performed. This was done by creating model paths from the query set, and deriving paths from the same query set. As an optimal solution with fitness $F = 1$ is available in this set-up, the competence of the search algorithm is being tested. No gaps were allowed in the benchmarking experiments.

In an initial experiment the first N_m gene products in the Lee et al. (2002) model graph were arranged (in random order) into a linear path consisting of N_m nodes, and the full set of 64 gene products was used as a query set. The same random order of gene products was used for all tested search algorithms. Appropriate parameter settings for the different algorithms were chosen by performing many smaller experiments, which are not reported here. In overall, parameters were chosen in a way so that high quality solutions are derived, while at the same time avoiding that an algorithm degenerates into another (simpler) algorithm, something which is explained in the analysis of the benchmark experiments. The maximum number of fitness evaluations was set to $2.5 \cdot 10^5$ for all search algorithms. The mutation probability was set according to path length, and it was found that it is appropriate to mutate approximately one gene product on average for paths of lengths less than 64, and on average two gene products for paths ≥ 64 gene products. Five different path lengths were tested, with mutation probabilities in brackets; 5 (0.2), 10 (0.1), 20 (0.05), 64 (0.031) and 211 (0.0095). The last probability is used in a later experiment using the Nilsson et al. (2006) dataset. Equal weight was given to the three sub-ontologies in the fitness function, i.e. $w_f = w_p = w_c = \frac{1}{3}$. Since all algorithms have stochastic elements, the performance may differ between runs with the same parameter settings. Therefore, 10 runs were performed for each algorithm. The algorithm specific settings are described in the following. For the EA, the following parameter settings were used: population size=10, crossover probability=0.8. For simulated annealing

(SA), the parameters were chosen as follows: start temperature $T = 0.001$, minimum temperature $T_{min} = 0.00001$, temperature decrease ratio $r = 0.9998158$, number of inner loop iterations $N = 10$. The stochastic hillclimber (SHC) parameter T was set to 0.0005. For the iterated hill-climber (HC), no restarts were used, maximum number of neighbour tries $M = 2.5 \cdot 10^5$ (which is also the maximum number of fitness evaluations allowed). Greedy search has no parameters except the sizes of the model path and query alphabet, which is part of the problem. Random search was only bounded by the maximum number of allowed fitness evaluations.

The results are shown in table 5.1, where the averages of maximum fitness, number of fitness evaluations and time elapsed, is shown for each combination of algorithm and model path length. It can be observed that the first four algorithms (EA, SA, SHC and HC) all reach maximum fitness for all model path lengths. The number of required fitness evaluations is fairly proportional to the number of gene products in the path. The parameter settings for EA, SA, and SHC are set to values that promote more local exploitation than global exploration. In the EA, a larger population than 10 individuals resulted in a larger number of fitness evaluations without an increased quality of the best solutions. By reducing the number of individuals even further, the performance of the EA will increase, but will essentially render the algorithm into an ordinary hill-climber apart from the crossover operator. Furthermore, the fitness variance of the EA population is small even long before the maximum fitness has been reached, which may indicate that the query set of gene products has a rather high semantic homogeneity or that there are subtle differences in terms of semantic similarity score over different solutions. Different settings promoting the acceptance of bad moves to a larger extent were also tested for SA and SHC, but it was found that there was no or little benefit of allowing worse solutions in the beginning of the search for this particular problem. The amount of elapsed time for the first four algorithms for path lengths greater than 20 is higher than it could be if the implementation was more efficient. As mentioned earlier, the variation operators for the first four algorithms have linear complexity, but it was discovered that a language construct in the inner loop for checking the presence of an item in an array also had linear complexity, rather than constant complexity like for a hash table. This

leads to exagerrated running times for long paths. The possible execution time is approximately one third or less of the specified running time when paths are longer than 20 gene products. However, the relative change in running time is still relevant when comparing algorithms for the same path length. It is obvious that the EA is the least efficient of the first four algorithms, and that the other three exhibit similar performance in terms of number of fitness evaluations. When studying the results of greedy search, it is obvious that it is much quicker than the first four algorithms, only needing approximately one tenth as many calculation steps as hill-climbing. Since GS does not operate on complete solutions, it does not use the ordinary evaluation function with linear complexity. This is the reason it is so much faster. However, GS does not manage to produce an optimal result for paths containing 64 gene products. Random search is shown merely as a baseline result, indicating what can be expected by just guessing. The maximum amount of fitness evaluations ($2.5 \cdot 10^5$) was used for RS, but the best solution was found after the number of evaluations shown in the table, and there was no improvement for the remaining fitness evaluations. Enumeration is not a feasible alternative, since it would take anything from approximately 10 hours to 10^{77} years depending on the model path length. It is assumed that one evaluation would take approximately as much time as one evaluation in the hill-climbing algorithm.

As a complement to table 5.1, the progress of the four first search algorithms is illustrated using fitness diagrams for model path lengths 10 (figure 5.5) and 64 (figure 5.6). The curves show the average fitness value of the 10 runs as a function of number of fitness evaluations. It can be observed in 5.5 that SA, SHC and HC have very similar behavior in fitness progression, and that the EA needs more iterations to converge to an optimal solution. It should be noted that a fitness of 0.95 is reached early in the search for SA, SHC and HC, approximately at half the number of fitness evaluations required to reach an optimal solution (fitness=1.0). The trends are the same for a model path length of 64, where SA, SHC and HC converge considerably quicker then the EA. A fitness of 0.95 is for SA, SHC and HC reached after about one third of the evaluations needed to find an optimal solution (see figure 5.7, which is a zoomed in version of figure 5.6).

Table 5.1: Benchmark results when using the Lee et al. (2002) *S. cerevisiae* query set of 64 gene products. N_m is model path length tested. F is the average of maximum fitness according to equation 5.1 when optimising using a specific algorithm ten times, i is average number of fitness evaluations and t is average execution time in seconds (if nothing else is stated). The algorithms used are evolutionary algorithm(EA), simulated annealing(SA), stochastic hill-climber(SHC), iterated hill-climber(HC), greedy search(GS), random search(RS), and enumeration ($Enum$). For enumeration, the average number of evaluations required to find the optimal solution by enumeration is shown (result from equation 5.4 divided by two).

N_m	EA	SA	SHC	HC	GS	RS	$Enum$
5	F=1.00 i=$3.28 \cdot 10^3$ t=0.31	F=1.00 i=$3.00 \cdot 10^3$ t=0.19	F=1.00 i=$2.63 \cdot 10^3$ t=0.17	F=1.00 i=$3.20 \cdot 10^3$ t=0.20	F=1.00 i=$3.1 \cdot 10^2$ t=0.002	F=0.78 i=$4.45 \cdot 10^4$ t=3.52	F=1.00 i=$4.6 \cdot 10^8$ t=9.73h
10	F=1.00 i=$1.03 \cdot 10^4$ t=1.33	F=1.00 i=$5.97 \cdot 10^3$ t=0.53	F=1.00 i=$6.40 \cdot 10^3$ t=0.57	F=1.00 i=$5.8 \cdot 10^3$ t=0.50	F=1.00 i=$5.96 \cdot 10^2$ t=0.003	F=0.72 i=$1.63 \cdot 10^4$ t=1.41	F=1.00 i=$2.7 \cdot 10^{17}$ t=10^5y
20	F=1.00 i=$2.41 \cdot 10^4$ t=5.02	F=1.00 i=$1.22 \cdot 10^4$ t=1.75	F=1.00 i=$1.71 \cdot 10^4$ t=2.43	F=1.00 i=$1.35 \cdot 10^4$ t=1.91	F=1.00 i=$1.09 \cdot 10^3$ t=0.005	F=0.67 i=$3.22 \cdot 10^4$ t=4.01	F=1.00 i=$2.4 \cdot 10^{34}$ t=10^{22}y
64	F=1.00 i=$6.31 \cdot 10^4$ t=74.88	F=1.00 i=$4.41 \cdot 10^4$ t=39.26	F=1.00 i=$3.39 \cdot 10^4$ t=30.03	F=1.00 i=$2.27 \cdot 10^4$ t=20.19	F=0.95 i=$2.08 \cdot 10^3$ t=0.011	F=0.60 i=$2.49 \cdot 10^4$ t=7.01	F=1.00 i=$6.3 \cdot 10^{88}$ t=10^{77}y

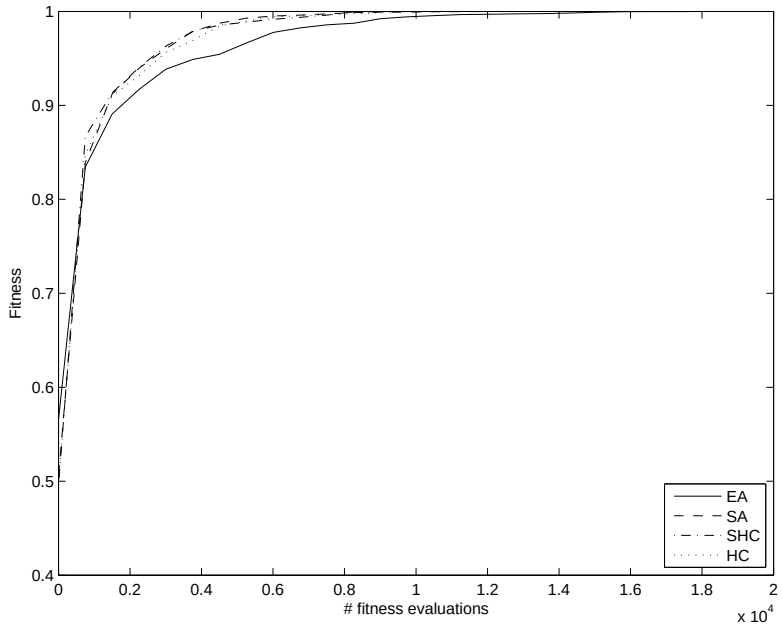


Figure 5.5: Average fitness as function of number of fitness calculations when using the Lee et al. (2002) *S. cerevisiae* query set of 64 gene products, and a model path of length 10 from the same set. Results are shown for EA (solid line), SA (dashed line), SHC (dash-dotted line) and HC (dotted line).

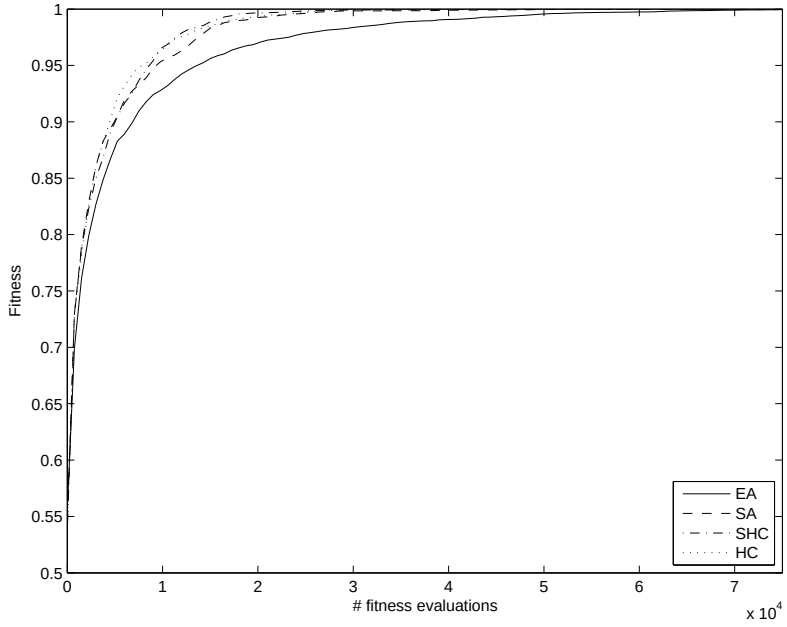


Figure 5.6: Average fitness as function of number of fitness calculations when using the Lee et al. (2002) *S. cerevisiae* query set of 64 gene products, and a model path of length 64 from the same set.

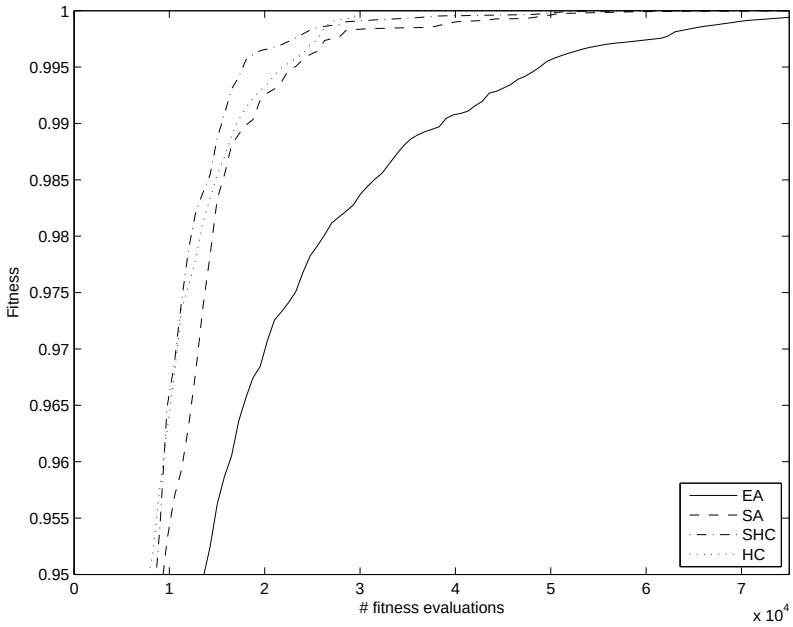


Figure 5.7: Zoomed in version of figure 5.6.

An equivalent experiment was performed using the *M. musculus* transgenic data set (Nilsson et al. 2006) containing 211 gene products. The same EA parameters were used as in the initial experiment using the model graph in Lee et al. (2002). Table 5.2 shows the results from this experiment. An initial observation is that all of the four first algorithms except SHC reach maximum fitness but in a larger number of iterations compared to the previous experiment. The maximum number of allowed fitness evaluations was increased to $7.5 \cdot 10^5$. By slightly adjusting the T parameter of SHC, it will probably be possible to achieve maximum fitness. One reason for the better performance on the first dataset is probably that it is more semantically homogeneous than the second dataset, i.e. the gene products are more semantically similar to each other because they are all related to transcription. In the second dataset, gene products are more different with respect to their GO annotations, making the optimisation task harder. Furthermore, the query set is more than three times larger in the second case. To sum up, the benchmark results indicate that the GO annotations of gene products in both model graph and query set clearly affect the qualitative performance of the search algorithms. Just as in the previous experiment, the EA is the least efficient of the first four algorithms, but the other three derive optimal solu-

tions with similar efficiency. Greedy search is optimal up to a model path length of 20, however slightly suboptimal for path lengths of 64 and 211. In fact, all of the first four algorithms are significantly better qualitatively than GS at the 99% level using a Student's t-test assuming different variances of the distributions for the two groups of replicate runs under comparison. Hence, given enough iterations and appropriate parameter settings, all of the first four search algorithms will perform better than greedy search if paths are long enough. Like for the Lee et al. (2002) data set, random search performs worse as the model sequence length increases. Enumeration is even more infeasible than for the earlier dataset, with an estimated execution time of 176 days for length 5 paths, which can be compared with approximately 10 hours for the Lee et al. (2002) data set. This illustrates the combinatorial explosion in number of possible solutions when the size of the query set increases.

Like for the Lee et al. (2002) data set, the progress of the four first search algorithms is shown for model path lengths 10 (figures 5.8 and 5.9), 64 (figure 5.10), and 211 (figure 5.11). Interestingly, the solution quality for the EA improves as quickly as the other algorithms for a path length of 10, but is like earlier slower in its progress compared to the other algorithms when the path length is 64. For a length of 211, the progress is generally slower for all algorithms. HC progresses most quickly, and the EA is approximately on par with SA, whereas SHC is slowest and also converges to a suboptimal solution.

The reason for using the same dataset both for model path and query set is that the optimal solution is known, yielding a fitness value of 1. However, in a real application of GOSAM, the gene products in the model path and query set often come from different data sets. In that case, the optimum fitness is not known. For this reason we have performed an additional experiment where the Lee et al. (2002) *S. cerevisiae* data set is used to create model paths and the *M. musculus* transgenic data set of Nilsson et al. (2006) is used as query set. The relative difference in performance between the algorithms is studied, rather than the absolute performance. The results are shown in table 5.3. For a model path length of 10 gene products it can be observed that there is a difference in quality in the third decimal, and all algorithms are in fact significantly better than greedy search at the 98% level using the same t-test as before.

Table 5.2: Benchmark results when using the *M. musculus* transgenic query set (Nilsson et al. 2006) of 211 gene products. For descriptions, see table 5.1.

N_m	*EA*	*SA*	*SHC*	*HC*	*GS*	*RS*	*Enum*
5	F=1.00 i=$9.46 \cdot 10^3$ t=1.13	F=1.00 i=$7.90 \cdot 10^3$ t=0.62	F=1.00 i=$7.54 \cdot 10^3$ t=0.59	F=1.00 i=$5.91 \cdot 10^3$ t=0.45	F=1.00 i=$1.05 \cdot 10^3$ t=0.005	F=0.68 i=$2.51 \cdot 10^4$ t=3.04	F=1.00 i=$2.0 \cdot 10^{11}$ t=176d
10	F=1.00 i=$3.06 \cdot 10^4$ t=4.81	F=1.00 i=$3.14 \cdot 10^4$ t=3.33	F=1.00 i=$3.02 \cdot 10^4$ t=3.14	F=1.00 i=$1.86 \cdot 10^4$ t=1.92	F=1.00 i=$2.07 \cdot 10^3$ t=0.01	F=0.45 i=$7.31 \cdot 10^4$ t=9.86	F=1.00 i=$7.0 \cdot 10^{22}$ t=10^{11}y
20	F=1.00 i=$1.06 \cdot 10^5$ t=24.57	F=1.00 i=$5.56 \cdot 10^4$ t=8.95	F=1.00 i=$4.88 \cdot 10^4$ t=7.78	F=1.00 i=$4.15 \cdot 10^4$ t=6.59	F=1.00 i=$4.03 \cdot 10^3$ t=0.02	F=0.37 i=$1.99 \cdot 10^4$ t=3.76	F=1.00 i=$1.7 \cdot 10^{29}$ t=10^{17}y
64	F=1.00 i=$6.74 \cdot 10^5$ t=494.13	F=1.00 i=$1.98 \cdot 10^5$ t=86.13	F=1.00 i=$2.18 \cdot 10^5$ t=95.09	F=1.00 i=$2.03 \cdot 10^5$ t=90.29	F=0.99 i=$1.15 \cdot 10^4$ t=0.06	F=0.31 i=$1.90 \cdot 10^4$ t=8.04	F=1.00 i=$6.0 \cdot 10^{45}$ t=10^{34}y
211	F=1.00 i=$6.39 \cdot 10^5$ t=5636	F=1.00 i=$6.29 \cdot 10^5$ t=3969	F=0.96 i=$6.11 \cdot 10^5$ t=3945	F=1.00 i=$3.33 \cdot 10^5$ t=2111	F=0.95 i=$2.24 \cdot 10^4$ t=0.12	F=0.28 i=$1.57 \cdot 10^4$ t=19.34	F=1.00 i=$3.6 \cdot 10^{118}$ t=10^{106}y

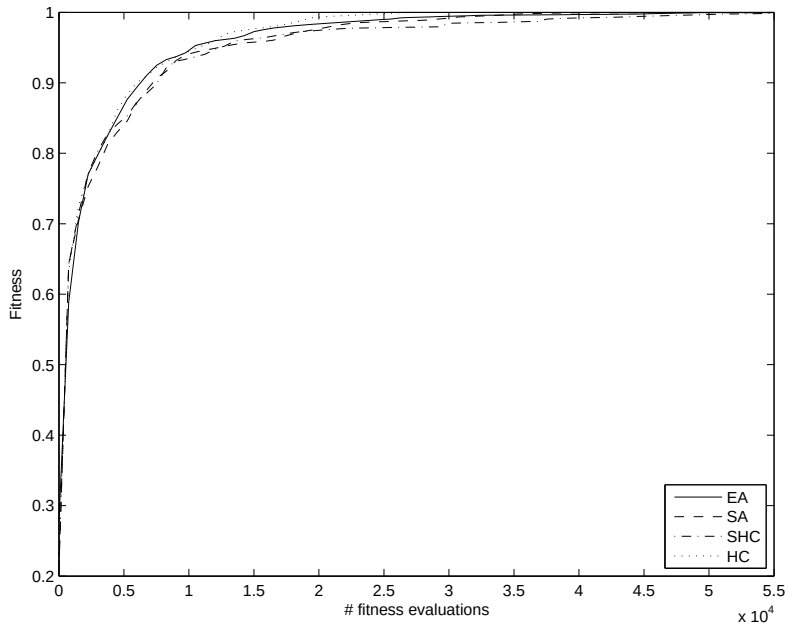


Figure 5.8: Average fitness as function of number of fitness calculations when using the *M. musculus* transgenic query set (Nilsson et al. 2006) of 211 gene products, and a model path of length 10 from the same set.

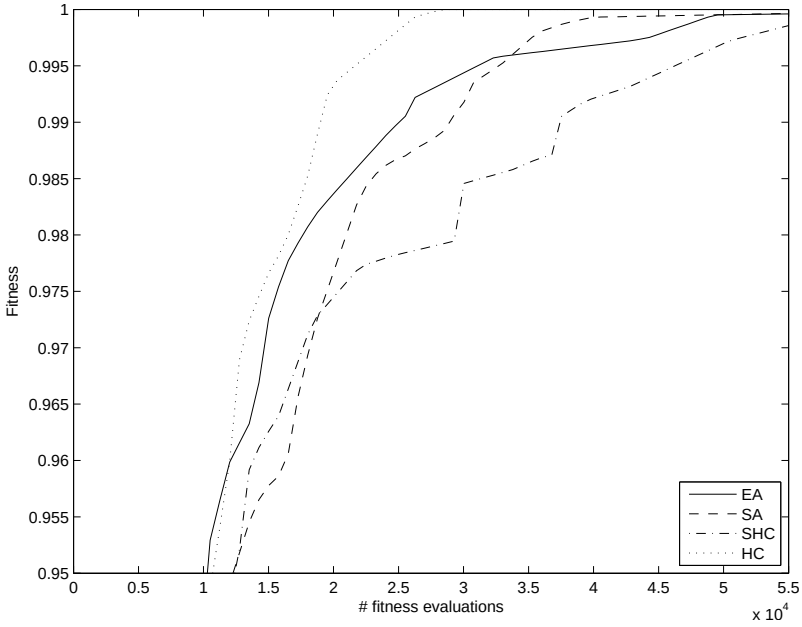


Figure 5.9: Zoomed in version of figure 5.8.

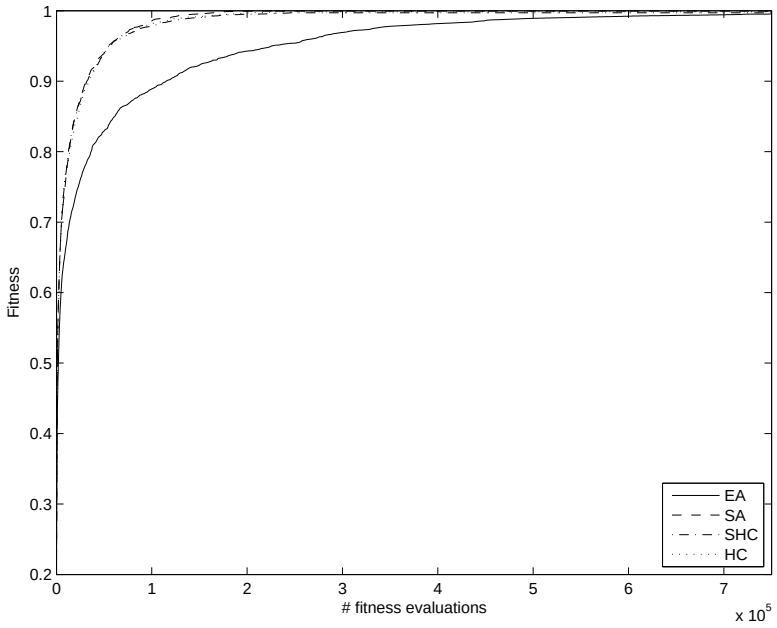


Figure 5.10: Average fitness as function of number of fitness calculations when using the *M. musculus* transgenic query set (Nilsson et al. 2006) of 211 gene products, and a model path of length 64 from the same set.

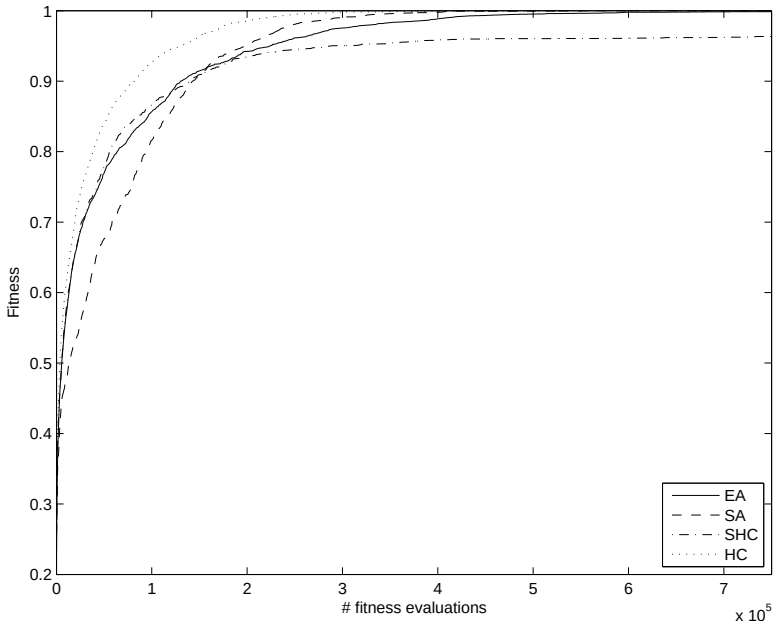


Figure 5.11: Average fitness as function of number of fitness calculations when using the *M. musculus* transgenic query set (Nilsson et al. 2006) of 211 gene products, and a model path of length 211 from the same set.

The difference in fitness between HC, EA, SA and SHC, is not significant. When the path length is increased to 64, the quality of the first four algorithms is better in the second decimal at a statistical significance level of 99.9%. SA is slightly better than EA, SHC and HC at a significance level of 99.9%. The EA and HC are better than SHC at the same level of significance.

When studying the fitness progress of the first four algorithms for path lengths 10 (figure 5.12) and 64 (figure 5.13), it is evident that the fitness rises quickly to 0.6 in the first case and slowly converges to approximately 0.65. The EA has a slower rate of convergence. For a path length of 64, HC has the fastest initial growth in fitness followed by SHC. However, SA surpasses HC in the long run. SHC arrives at a suboptimal solution and EA has the slowest overall growth in fitness.

Table 5.3: Benchmark results when using the *M. musculus* transgenic data set (Nilsson et al. 2006) of 211 gene products as query set, and the Lee et al. (2002) *S. cerevisiae* data set of 64 gene products as model set. This test was done in order to assess the greedy search algorithm. For descriptions, see table 5.1.

N_m	EA	SA	SHC	HC	GS
10	F=0.658	F=0.658	F=0.658	F=0.656	F=0.652
	i=$7.17 \cdot 10^4$	i=$4.84 \cdot 10^4$	i=$4.08 \cdot 10^4$	i=$3.90 \cdot 10^4$	i=$2.07 \cdot 10^3$
	t=11.4	t=5.47	t=4.60	t=4.39	t=$1.06 \cdot 10^{-2}$
64	F=0.576	F=0.579	F=0.566	F=0.576	F=0.546
	i=$6.37 \cdot 10^5$	i=$6.15 \cdot 10^5$	i=$3.97 \cdot 10^5$	i=$3.09 \cdot 10^5$	i=$1.15 \cdot 10^4$
	t=463.15	t=268.62	t=178.85	t=141.14	t=$6.23 \cdot 10^{-2}$

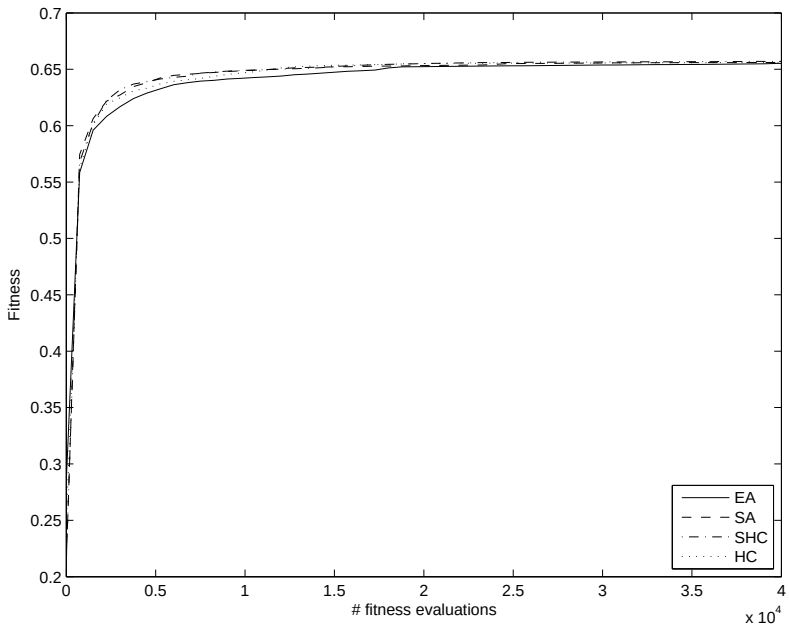


Figure 5.12: Average fitness as function of number of fitness calculations when using the *M. musculus* transgenic data set (Nilsson et al. 2006) of 211 gene products as query set, and a model path of 10 gene products from the Lee et al. (2002) *S. cerevisiae* data set.

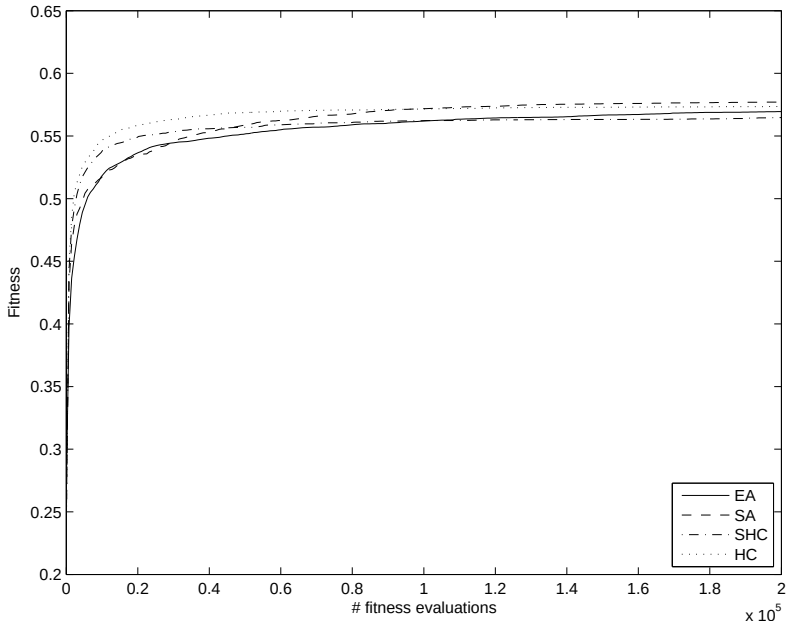


Figure 5.13: Average fitness as function of number of fitness calculations when using the *M. musculus* transgenic data set (Nilsson et al. 2006) of 211 gene products as query set, and a model path of 64 gene products from the Lee et al. (2002) *S. cerevisiae* data set.

5.3.4 Cross-species experiments

For the cross-species experiments only the EA was used as search algorithm, and the same parameter settings were used as in the benchmark experiments. 100 randomised model graphs were used for calculation of statistical significance. In the first experiment, the Lee et al. (2002) model and both *M. musculus* query sets were used. Figure 5.14 shows the percentage of paths for which alignments with significant p-value were found, as a function of p-value threshold. This test was performed both for the 105 super-paths of the model graph, and for the complete set of 204 possible paths of all lengths. For both query sets it can be observed that significant alignments were found for only a few of the paths with threshold $p \leq 0.02$. Furthermore, significant alignments are found for a larger proportion of super-paths compared to the case with all possible paths. One reason for this effect is that many of the latter mentioned paths are short and therefore less likely to be significant. Given that this is a cross-species scenario, comparing two distantly related species, and considering the rather small numbers of gene products in the two datasets, we do not find it surprising that

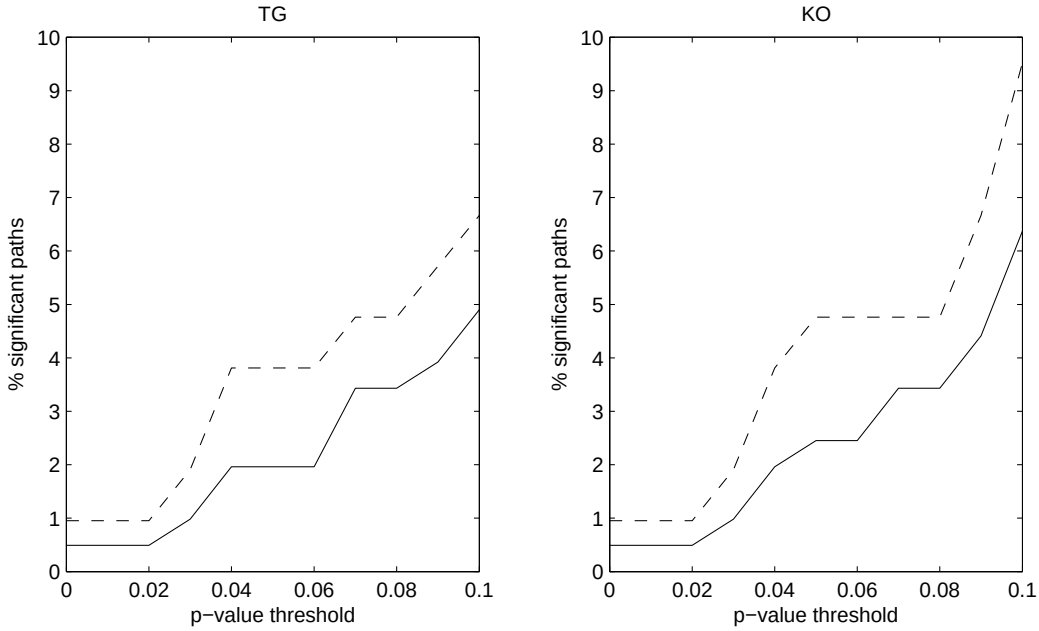


Figure 5.14: Percentage of paths for which significant alignments are found as a function of p-value threshold for the transgenic dataset (TG) and knock-out dataset (KO). Solid lines represent the case when all possible paths were used and dashed lines represent the case with super-paths.

significant alignments are found only for a small proportion of the paths.

To illustrate the output of GOSAM, we here show examples of putative path alignments derived using the knock-out query set of differentially expressed genes for *M. musculus* and the super-paths extracted from the Lee et al. model graph. An example is the query path "NFIX → AKAP8 → STAT5B", which resulted in the alignment shown in table 5.4. This alignment has $F = 0.72$ and $p = 0$. Q is the derived path, and M is the model path. F shows the GO molecular function meta-alignment, where each identifier represents the minimum subsumer GO term for the two gene products under comparison. The corresponding information for biological process and cellular component is shown in the rows labelled by P and C. For example, in the molecular function meta-alignment, gene products NFIX1 and MBP1 have the minimum subsumer "transcription factor activity" (GO:0003700), and are both involved in "DNA replication" (GO:0006260) and expressed in the nucleus (GO:0005634). In the second position, both gene products have the function "chromatin binding" (GO:0003682) and are involved in "chromosome organization and biogenesis" (GO:0051276) and are

Table 5.4: Example alignment obtained when using the transgenic data set as query set and the Lee et al. graph as model. For explanations, see text.

Alignment					
Q:	NFIX	→	AKAP8	→	STAT5B
M:	MBP1	→	ABF1	→	STP1
Meta-alignment					
F:	GO:0003700	→	GO:0003682	→	GO:0003700
P:	GO:0006260	→	GO:0051276	→	GO:0045944
C:	GO:0005634	→	GO:0000790	→	GO:0005634

expressed in the "nuclear chromatin" (GO:0000790). In the third position, the two gene products STAT5B and STP1 share the function "transcription factor activity", the biological process "positive regulation of transcription from RNA polymerase" and the cellular location "nucleus". Thus, the alignment clearly indicates that these are homologous paths of transcriptional regulation, which have been preserved between yeast and mouse.

Interestingly, it is unlikely that this similarity would have been found by using a traditional approach based on sequence homology. We found that the average sequence identity between the three pairs of amino acid sequences was only 14.4%. Furthermore, for each one of the three query sequences another match with slightly higher sequence identity (16.1% on average) could be found by BLAST-searching the *M. musculus* subset of REFSEQ (Pruitt et al. 2007). When studying the annotations of these "closer homologs", we found that their function was much less similar and that their cellular location included cytoplasm and membrane, which clearly indicates that a sequence-based approach would have produced spurious hits for this query path. In the example alignment in table 5.4, it can also be observed that despite the obviously high similarity between the two paths, they do not have exactly the same annotations throughout the whole alignment. At position two, the shared biological process annotation term is "chromosome organization and biogenesis", since this is the minimum subsumer of the terms found in the annotation of AKAP8 and ABF1. However, looking at the concrete annotation of the gene products, we find that AKAP8

Table 5.5: Example alignment obtained when using the transgenic data as query set and the Lee et al. graph as model. GO term codes (shown without "GO:" and initial zeros) have the following interpretations: 3713: transcription coactivator activity, 3700: transcription factor activity, 16563: transcriptional activator activity, 8237: metallopeptidase activity, 16564: transcriptional repressor activity, 6366: transcription from RNA polymerase II promoter, 7049: cell cycle, 6357: regulation of transcription from RNA polymerase II promoter, 6508: proteolysis, 122: negative regulation of transcription from RNA polymerase II promoter, 5634: nucleus, 16021: integral to membrane, 5694: chromosome.

Alignment									
Q:	MEF2C	→	NR2F6	→	NRBF2	→	AFG3L2	→	TRIM28
M:	SWI6	→	SWI4	→	NDD1	→	ACE2	→	SFL1
Meta-alignment									
F:	3713	→	3700	→	16563	→	8237	→	16564
P:	6366	→	7049	→	6357	→	6508	→	122
C:	5634	→	5634	→	5634	→	16021	→	5694

is annotated with "mitotic chromosome condensation" and that ABF1 is annotated with "nucleotide-excision repair / DNA damage recognition". This difference accounts for the fitness score being $F = 0.72$, rather than $F = 1$, but also demonstrates that relatively modest fitness scores may correspond to high quality alignments between closely homologous paths. Another example of a significant alignment ($F = 0.73$, $p = 0.01$), obtained when using the transgenic query set, is shown in table 5.5.

In the second experiment, we used as models the cell cycle and MAPK pathways for *S. cerevisiae*, both from KEGG. In each case, the *M. musculus* transgenic dataset was used as query set, but with the addition of all gene products from the corresponding pathway (cell cycle or MAPK) for *M. musculus*. The percentage of significant paths as a function of p-value is shown in figure 5.15. This test was performed for both pathways, and also both for the super-paths and the complete set of possible paths. For both model pathways it can be observed that significant alignments were derived for a large proportion of the paths, even for such a conservative significance threshold as $p = 0$. One reason for this is probably that paths on average are consid-

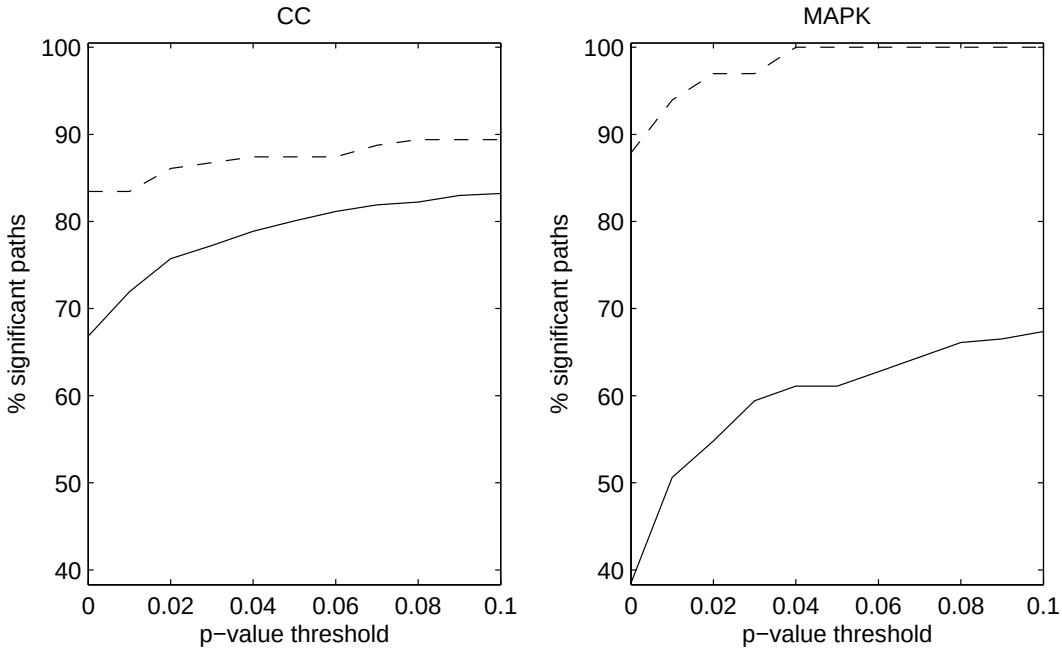


Figure 5.15: Percentage of paths for which significant alignments are found as a function of p-value threshold for the cell cycle pathway (CC) and MAPK pathway (MAPK). Solid lines represent the case when all possible paths were used and dashed lines represent the case with super-paths.

erably longer for the cell cycle and MAPK pathways than for those from the Lee et al. (2002) model. Optimised alignments containing long paths are less likely to appear by chance when using the Maslov and Sneppen graph randomisation procedure. Furthermore, the cell cycle and MAPK pathways are less semantically homogeneous compared to the Lee et al. (2002) model, which only contains gene products related to transcription. Lower semantic homogeneity would probably yield a larger number of significant paths. Another reason may be that the scores of alignments are generally higher since the transgenic dataset has been "injected" with all the gene products from the corresponding *M. musculus* pathway (cell cycle or MAPK). In fact, all optimised alignments contain query paths where all gene products are from the corresponding *M. musculus* pathway, i.e. no gene products were selected from the original transgenic dataset. Gene products from the corresponding pathway are exclusively selected also if the transgenic dataset is replaced with the knockout dataset.

An example of a significant alignment ($F = 0.86$, $p = 0$) is shown in table 5.6. This alignment was derived using the cell cycle pathway as model. In the model path,

Table 5.6: Example alignment obtained when using the *M. musculus* transgenic data as query set, with the cell cycle gene products for the same organism added. The cell cycle pathway for *S. cerevisiae* was used as model. A pipe sign "|" separates GO terms resulting in equal score. GO term codes have the following interpretations: 19210: kinase inhibitor activity, 4674: protein serine/threonine kinase activity, 16563: transcriptional activator activity, 79: regulation of cyclin dependent protein kinase activity, 51320: S phase, 84: S phase of mitotic cell cycle, 6357: regulation of transcription from RNA polymerase II promoter, 5634: nucleus.

Alignment					
Q:	CDKN1B	→	DBF4	→	SMAD4
M:	SIC1	→	CDC28	→	SWI5
Meta-alignment					
F:	19210	→	4674	→	16563
P:	79	→	51320|84	→	6357
C:	5634	→	5634	→	5634

SIC1 inhibits the cyclin-dependent protein kinase CDC28. CDC28 in turn phosphorylates and inhibits SWI5, which is a transcriptional activator. The corresponding *M. musculus* gene products in the alignment are not connected in the KEGG cell cycle pathway for *M. musculus*. It should be emphasised that the cell cycle pathways for *S. cerevisiae* and *M. musculus* have similar regulatory mechanisms, but the pathway topologies are quite different. This may explain why the derived query path is not part of the *M. musculus* cell cycle pathway, even though close gene product homologs were found. An interesting observation is that SWI5 expresses SIC1 according to KEGG, and the same regulation is present between SMAD4 and CDKN1B in the *M. musculus* cell cycle pathway. Thus, the same relationship between the last and first position in the alignment is present in both paths.

Another example of a significant alignment ($F = 0.80$, $p = 0$) is shown in table 5.7. This alignment was derived using the MAPK pathway as model. The model path represents a part of the high osmolarity induced sub-pathway where STE20 is a signal-inducing kinase which phosphorylates STE11, which is a signal-transducing

Table 5.7: Example alignment obtained when using the *M. musculus* transgenic data as query set, with the gene products of the MAPK pathway for the same organism added. The MAPK pathway for *S. cerevisiae* was used as model. GO term codes have the following interpretations: 4674: protein serine/threonine kinase activity, 4709: MAP kinase kinase kinase activity, 5076: receptor signaling protein serine/threonine kinase signaling protein activity, 5078: MAP-kinase scaffold activity, 4707: MAP kinase activity, 19953: sexual reproduction, 6468: protein amino acid phosphorylation, 187: activation of MAPK activity, 43406: positive regulation of MAPK activity, 42995: cell projection, 5737: cytoplasm, 5634: nucleus.

Alignment							
Q:	AKT1	→	B230120H23RIK	→	MAPK8IP3	→	MAPK1
M:	STE20	→	STE11	→	PBS2	→	HOG1
Meta-alignment							
F:	4674	→	4709	→	5076|5078	→	4707
P:	19953	→	6468	→	187|43406	→	6468
C:	42995	→	5737	→	5737	→	5634

MEK-kinase active during the MAPKKK phase of the MAPK pathway. STE11 in turn phosphorylates PBS2, a kinase active in the MAPKK phase. PBS2 finally phosphorylates HOG1, which is a another kinase operating during the MAPK phase. The *M. musculus* gene products in the query path are not connected in the *M. musculus* version of the MAPK pathway, but gene products in positions two through four appear in the correct phases in the pathway (MAPKKK, MAPKK and MAPK). As for the cell cycle pathway, the MAPK pathways for *S. cerevisiae* and *M. musculus* have similar mechanisms, but are rather different in their topologies.

5.4 Discussion

We have developed GOSAM, a method which uses a known pathway graph as a model from which model paths are extracted, and a search algorithm to derive putative paths that are semantically similar to these model paths using a set of experimentally

determined, or by other means derived, gene products. We have evaluated different kinds of search algorithms and tested the performance of the method on example datasets, and shown by examples the output produced by the method.

To sum up the experiments on search algorithms, the greedy search algorithm is superior in terms of efficiency due to a lower time complexity compared to the other algorithms. However, GS is slightly worse in terms of solution quality especially for paths with more than 20 gene products, and even for paths of length 10 in the final cross-species experiment. An evolutionary algorithm seems to be least efficient for this particular problem. The iterated hill-climber (used without restarts) seems to be the simplest and most efficient algorithm of the ones tested to derive optimal solutions in the GOSAM method. Its performance is similar to SHC and SA, but since Ockham's razor suggests the simplest approach if the performance of methods is equally good, the iterated hill-climber (as used here in a simplified manner without random restarts) arguably is the appropriate choice. It should also be mentioned again that the parameters of the EA, SA, and SHC algorithms were set so that the behaviour features considerably more local exploitation than global exploration, and therefore resembles hill-climbing search. A larger degree of global exploration was not found successful neither in terms of convergence speed nor solution quality.

As mentioned in the section on complexity, the execution time of the path optimisation procedure in GOSAM depends on many parameters, e.g. search algorithm, the number of model paths, the model path length, and the number of randomised models used in the significance calculation. To optimise the alignment of one path containing a reasonable number of gene products (< 10) is in the current implementation done in a matter of seconds on a modern PC with a processor speed of 3 GHz and 1 GB of RAM.

Currently, the optimisation procedure has the restriction that a gene product can only appear once in a path. This restriction is reasonable for regulatory pathways, but may not be desirable when metabolic pathways are studied. The method could be adapted for this scenario by allowing each gene product to appear at several positions in a path. This modification would also require that variation operators are replaced or modified.

The method allows the user to set different weights for the different GO sub-ontologies, although we have only used $w_f = w_p = w_c = \frac{1}{3}$ in the presented evaluations. Setting different weights for the different types of annotation can be useful since the certainty of different types of annotation can vary a lot for different organisms. If it is found that a large proportion of the annotation in a particular sub-ontology is, for example, inferred by homology, rather than based on experimental evidence, then it may be desirable to give this type of annotation a lower impact in the alignment optimisation process.

Chapter 6

Thesis conclusions

6.1 Summary and comparison of methods

In this thesis three related methods have been developed for semantic comparison of biological pathways. The methods have several traits in common. Biological pathways are being analysed in all cases. GO, gene product annotation databases, and information theory are used in all methods. Furthermore, the concept of generalisation using GO categories of gene products is used throughout. Methods also aim to derive, in some sense, biologically plausible results. There are also some major differences between the methods; GOTEM uses binary interactions between gene products in pathways, whereas GOSAP and GOSAM use paths of gene products. Additionally, GOTEM derives structures of knowledge (templates) from a model pathway prior to comparison with a query pathway, whereas GOSAP and GOSAM derive structures of knowledge (alignments) during the comparison itself. Furthermore, GOSAP and GOSAM use statistical tests to assess the significance of alignments whereas GOTEM uses a more arbitrary score threshold approach to template match assessment. Finally, GOTEM and GOSAP assume a query graph, whereas GOSAM only assumes a query set of gene products.

In the introduction of this thesis, seven contributions of the work are listed. These contributions have been motivated and justified in the chapters for the proposed methods GOTEM (chapter 3), GOSAP (chapter 4), and GOSAM (chapter 5). In more detail, the first contribution encompasses the GOTEM method itself, and is justified

in the methods section (3.2) and by the results reported in section 3.3, using data from regulatory pathways in *S. cerevisiae* and *H. sapiens*. In particular, the results demonstrate that the method is able to filter out a large proportion of potentially implausible hypotheses, thus greatly improving the specificity of the regulatory network reconstruction process. Of course, there is potential for method improvements, which is discussed in section 6.2. The second contribution concerns the GOSAP method itself, and is justified in the methods section (4.2) and by the results reported in section 4.3, using data from regulatory- and metabolic pathways in *S. cerevisiae*, *E. coli*, *H. sapiens* and *M. musculus*. The main novelty of the method is that any kind of biological pathway where nodes are gene products can be aligned semantically, i.e. one is not restricted to enzyme-to-enzyme metabolic pathways. Results from application examples presented in section 4.3 demonstrate the generality of GOSAP and the range of application areas. As for GOTEM, there are ideas for improvements, which are discussed in section 6.2. The third contribution suggests that the sensitivity and specificity of the path alignment process can be improved by combining the function- , process- and component ontologies of GO, and this is justified by the empirical results in section 4.3.5, using the orthologous cell cycle pathways in KEGG for *H. sapiens* and *M. musculus*. Most importantly, these experiments show that the ROC area increases when sub-ontologies are combined, which suggests a better tradeoff between sensitivity and specificity. Contribution number four encompasses novel results of how enzyme-to-enzyme paths in the metabolic pathways of *S. cerevisiae* and *E. coli* (as documented in the SGD (Saccharomyces Genome Database, www. yeastgenome.org)) are related semantically, which is justified by various empirical evidence in section 4.3.4. Most important result in the justification of the contribution is table 4.3 which shows 60 semantic metabolic pathway homologies between the two species. Examples from this table are visualised in figures 4.6 and 4.7. The fifth contribution concerns novel results of a comparison between semantic similarity and amino acid sequence similarity in the assessment of how well a path alignment algorithm can separate documented paths (true positives) from currently unknown paths (false positives), which is justified by the experiments reported in section 4.3.5, using the orthologous cell cycle pathways in KEGG for *H. sapiens* and *M. musculus*. The results indicate that it is benificial

to use semantic- and sequence similarity in combination, since the measures derive complementary sets of currently unknown paths. Contribution six encompasses the GOSAM method itself, and is justified in the methods section (5.2) and by the results reported in section 5.3, using various search algorithms and data from regulatory and signalling pathways in *S. cerevisiae* and *M. musculus*. We have shown by examples the output of the method, but as for all methods proposed in this thesis, there has been no biological validation (in a lab) of any of the derived (hypothetical) results. The validation is, of course, important but is considered to be out of scope for this thesis. Just as for GOTEM and GOSAP, there is potential for method improvements, which is discussed in section 6.2. The final contribution (number seven) encompasses the novel results of how different search algorithms (random search, greedy search, iterated hill-climbing, stochastic hill-climbing, simulated annealing and an evolutionary algorithm) perform in the assembly of putative regulatory paths being semantically similar to paths in documented pathways. The contribution is justified by the experiments in section 5.3.3, using data from regulatory and signalling pathways in *S. cerevisiae* and *M. musculus*, which showed that greedy search is superior in terms of speed but most often suboptimal. For this reason iterated hill-climbing is considered to be the best choice in terms of solution quality for this particular problem.

It is argued that the methods proposed in this thesis will be useful to biologists in order to assess the biological plausibility of derived pathways, compare different pathways for semantic similarities, and to derive putative pathways that are semantically similar to documented biological pathways. To our knowledge, all methods contain novel elements, and will therefore extend the systems biology toolbox that biologists can use to make new biological discoveries.

6.2 Future work

There are, of course, additional sources of biological knowledge that could be incorporated into the methods in future versions. As mentioned in chapter 3, a way to get better accuracy for our GOTEM method could also be to incorporate databases containing transcription factor binding site information or protein interactions. This

is a more specific kind of knowledge that has been used by other researchers (e.g. Hartemink et al. (2002) and Haverty et al. (2004)), and that could potentially be combined with our GO-based approach. In chapter 4 we propose how GOSAP can be generalised to be even more useful in the same problem domain by adding additional similarity measures for gene products, such as sequence similarity and structure similarity. We also suggest the use of additional biological knowledge in the form of ChEBI in order to be able to include smaller molecules (metabolites) in metabolic pathways in the semantic similarity calculations. Extensions similar to those proposed for GOSAP would also apply to GOSAM.

GOSAP could in the future be extended to support multiple alignments, which would enable the study of more than two species or paths at the same time. Additionally, GOSAP and GOSAM could be re-worked to support more complex alignments, such as tree- or even graph based alignments.

It would be possible to extend the GOSAM method to allow each gene product to appear at several positions in a path in order to make the method more useful in the study of metabolic pathways. The modification requires that variation operators are replaced or modified.

An in-depth study of the impact of different GO evidence types would be useful input to all three proposed methods. Currently, all kinds of evidence is used in the calculations. If the study shows that some evidence types are less reliable, these can e.g. be assigned smaller weights in the semantic similarity calculations.

In the future it would be of interest to apply the methods to specific application areas that have not been covered in the present experiments in the thesis. One example is the work on endometrial cancer by Karlsson (2006), where microarray gene expression measurements have been performed in rat models, and analysed using methods such as differential analysis and clustering. This scenario could serve as a good application of the GOSAM method, where putative pathways can be assembled from sets of differentially expressed genes, and compared to known pathways in other kinds of cancer. Another example is the work by Olsson et al. (2006) on the parsing of biomedical texts (e.g. PubMed abstracts) in order to derive relationships and pathways between different kinds of entities (e.g. genes and proteins) in the texts.

The pathways derived from text analysis could be aligned with other pathways using GOSAP.

6.3 Final reflections

One general observation regarding methods for analysis of large scale biological data is that it has become more common in recent years to combine different kinds of data and information. Data- or information fusion is a research area that has not been elaborated on in this thesis so far, but a lot of research in bioinformatics and systems biology can be mapped to data fusion, including our proposed methods. Methods for data fusion combine data from several sensors and databases to achieve better accuracy and inferences that could be achieved by using any of the data sources alone (Hall and Llinas 1997). Data fusion was originally intended for military applications, where sensors usually referred to e.g. radar antennas, motion sensors, and optical sensors. Databases in this context can contain information on military objects such as aircraft, tanks, ships and troops. In biology a sensor could represent a microarray probe or some other means of measuring properties of biological entities. Databases in the biological domain can for example contain the sequence, structure, and different kinds of annotation for biological entities such as genes, gene products and small molecules. The methods proposed in this thesis are all examples of data fusion methods, where data from different kinds of databases are combined. Currently the databases used are GO, GO annotation databases and pathway databases. Data from sensors are not explicitly used in our proposed methods, but derived pathways can implicitly contain sensor data because pathways can be derived from e.g. microarray gene expression data. As already mentioned, GOSAP could for example be extended to combine data from additional sources such as sequence- and structure databases. GOTEM may benefit from the addition of transcription factor binding site information and protein interaction data. The JDL model is a classic model of data fusion, and contains the different processes in data fusion and how they are related (Hall and Llinas 2000). It is used to create an overview of the different parts of a data fusion project and to facilitate the communication between people from different disciplines within and

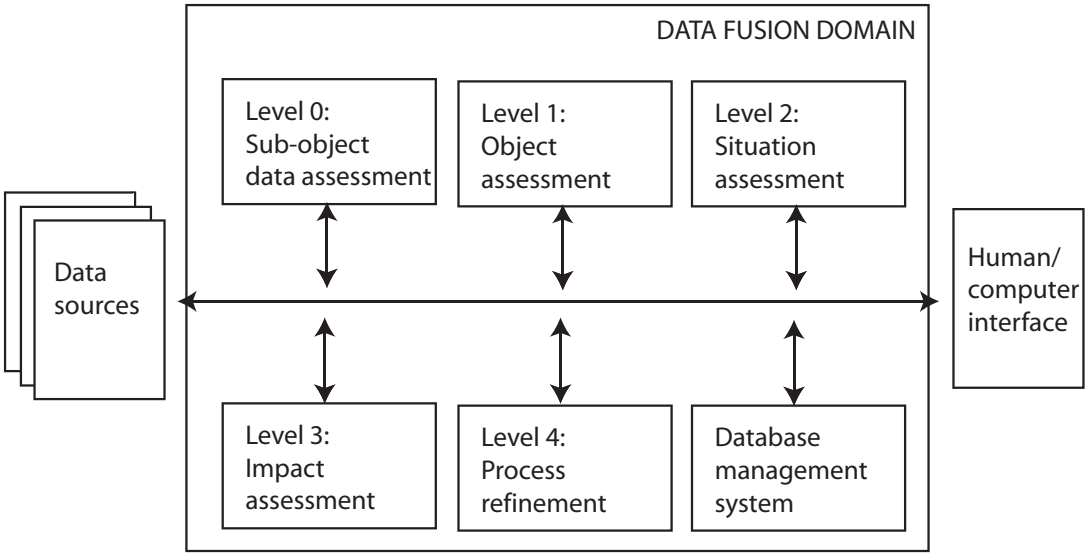


Figure 6.1: The extended version of the JDL model of data fusion.

outside a project. We have shown that this model is also applicable to bioinformatics (Synnergren et al. 2007). It is therefore also applicable to systems biology since bioinformatics can be classified as being part of systems biology. Figure 6.1 shows the extended version of the JDL model as described in Steinberg et al. (1999). To the left in the figure there are multiple data sources serving as input to the data fusion domain where increasingly complex processes, categorised into five levels, are used in different kinds of computational processing. There are also database management systems utilised in the representation and processing of databases. The results from the different processes are combined and eventually presented to a human or computer interface. The process level does not necessarily dictate the order of processing but rather the abstraction level and degree of sophistication. Level 0 (Sub-object data assessment) covers different kinds of data and signal processing necessary to perform on data sources. In bioinformatics this could for example encompass normalisation of microarray data, removal of noisy or low level data, and handling incomplete data sets. In our proposed methods, this process is not explicitly represented since the processing of low level biological data in not included in any of the methods. Level 1 (Object assessment) is the process where data is interpreted using different algorithms and objects in the domain of interest are identified. Examples of possible biological

objects are genes, proteins, cells and pathways. In our methods we do not identify any new genes or proteins, but the methods can identify new pathways. An example being GOSAM which assembles pathways semantically similar to documented pathways. The relations between relevant objects in the domain of interest are established at level 2 (situation assessment) in the JDL model. In bioinformatics, example relations are regulatory interactions between genes or proteins, clusters of co-expressed genes, different kinds of biological pathways, and relationships between pathways. In our methods we compare pathways and therefore establish a relationship between the pathways. In GOTEM the relationships derived are a number of high scoring regulatory templates connecting the two pathways under comparison. GOSAP relationships are represented as alignments between pathways under comparison, and we have also demonstrated an approach for establishing graphs where a node is a pathway and an edge represents that two connected pathways are related in at least one statistically significant GOSAP alignment. GOSAM relates a set of, possibly differentially expressed or otherwise related, genes to a documented pathway using an alignment derived by a search algorithm. Level 3 (impact assessment) is the most sophisticated level, and contains algorithms for estimation and prediction of the impact of future situations and plans. In the bioinformatics domain we extended the definition to also include the estimation and prediction of effects on a biological system when objects and relations are manipulated (Synnergren et al. 2007). This could for example encompass over-expression, silencing, or knock-out of genes in a microarray experiment. It could also involve the alteration of culturing conditions. The manipulation could be carried out using either simulations or experimental techniques. In our proposed methods this process is not explicitly represented. However, a pathway comparison may generate a hypothesis that needs to be tested in a lab, and may involve e.g. gene knockout experiments. At level 4 (Process refinement) the data acquisition and processing at all the other levels are adapted and refined in order to support the objectives of the data fusion project. In bioinformatics this can encompass the collection of interesting hypotheses and the design of novel experiments to test these hypotheses. In our proposed methods, putative relations between pathways are derived using different pathway comparison methods. The most promising putative relations can

be used to design more specific experiments, possibly also utilising reductionist experimental approaches. The results from the new experiments may in turn serve as input to new pathway comparisons. So, even if process refinement is not described in the method sections, it can certainly be used when the methods are applied in a project where several iterations, both experimental and computational, are allowed.

Bibliography

Aderem, A. (2005). Systems Biology: Its Practice and Challenges, *Cell* **121**: 511–513.

Adomavicius, G. (2002). *Expert-Driven Validation of Set-Based Data Mining Results*, PhD thesis, New York University.

Agrawal, R. and Srikant, R. (1994). Fast Algorithms for Mining Association Rules, *Proceedings of the 20th VLDB Conference, Santiago, Chile*, pp. 487–499.

Ahn, A. C., Tewari, M., Poon, C.-S. and Phillips, R. S. (2006). The Limits of Reductionism in Medicine: Could Systems Biology Offer an Alternative?, *PLoS Medicine* **3**: 709–713.

Akutsu, T., Miyano, S. and Kuhara, S. (1999). Identification of genetic networks from a small number of gene expression patterns under the boolean network model, *Pacific symposium on biocomputing*, pp. 17–28.

Alberts, B., Bray, D., Johnson, A., Lewis, J., Raff, M., Roberts, K. and Walter, P. (1998). *Essential Cell Biology - An Introduction to the Molecular Biology of the Cell*, Garland, New York.

Altschul, S., Gish, W., Miller, W., Myers, E. and Lipman, D. (1990). Basic local alignment search tool, *Journal of Molecular Biology* **215**: 403–410.

Alwin, J. C., Kemp, D. J. and Stark, G. R. (1977). Methods for detection of specific RNAs in agarose gels by transfer to diazobenzyloxymethyl-paper and hybridization with DNA probes, *Proc. Natl. Acad. Sci. USA* **74**: 5350–5354.

Ashburner, M., Ball, C. A., Blake, J. A., Botstein, D., Butler, H., Davis, A. P., Dolinski, K., Dwight, S. S., Eppig, J. T., Harris, M. A., Hill, D. P., Issel-Tarver, L., Kasarskis, A., Lewis, S., Matese, J. C., Richardson, J. E., Ringwald, M., Rubin, G. M. and Sherlock, G. (2000). Gene ontology: tool for the unification of biology, *Nature Genetics* **25**: 25–29.

Bansal, M., Belcastro, V., Ambesi-Impiombato, A. and diBernardo, D. (2007). How to infer gene networks from expression profiles, *Molecular Systems Biology* **3**: 78.

Berg, J. and Lässig, M. (2004). Local graph alignment and motif search in biological networks, *PNAS* **101**: 14689–14694.

Berg, J. and Lässig, M. (2006). Cross-species analysis of biological networks by Bayesian alignment, *PNAS* **103**: 10967–10972.

Bergmann-Sigurdsteinsdottir, G. (2004). *Learning gene interactions from gene expression data using dynamic Bayesian networks*, Master's thesis, University of Skövde.

Butcher, E. C., Berg, E. L. and Kunkel, E. J. (2004). Systems biology in drug discovery, *Nature biotechnology* **22**: 1253–1259.

Cakmak, A. and Ozsoyoglu, G. (2007). Mining biological networks for unknown pathways, *Bioinformatics* **23**: 2275–2783.

Caspi, R., Foerster, H., Fulcher, C. A., Kaipa, P., Krummenacker, M., Latendresse, M., Paley, S., Rhee, S. Y., Shearer, A. G., Tissier, C., Walk, T. C., Zhang, P. and Karp, P. D. (2008). The MetaCyc Database of metabolic pathways and enzymes and the BioCyc collection of Pathway/Genome Databases, *Nucleic Acids Research* **36**: D623–D631.

Chung, H.-J., Kim, M., Park, C. H., Kim, J. and Kim, J.-H. (2004). Arrayxpath: mapping and visualizing microarray gene-expression data with integrated biological pathway resources using scalable vector graphics, *Nucleic Acids Research* **32**: W460–W464.

Conover, W. J. (1971). *Practical nonparametric statistics*, Wiley, New York.

Crick, F. (1970). Central Dogma of Molecular Biology, *Nature* **227**: 561–563.

Dahlquist, K. D., Salomonis, N., Vranizan, K., Lawlor, S. C. and Conklin, B. R. (2002). GenMAPP, a new tool for viewing and analyzing microarray data on biological pathways, *Nature Genetics* **31**: 19–20.

Dandekar, T., Schuster, A., Snel, B., Huynen, M. and Bork, P. (1999). Pathway alignment: application to the comparative analysis of glycolytic enzymes, *Biochemistry Journal* **343**: 115–124.

Dayhoff, M., Schwartz, R. M., C., B. and Orcutt (1978). *Atlas of protein sequence and structure*, Vol. 5, National Biomedical Research Foundation, Silver Spring, Maryland, chapter A model of evolutionary change in proteins, pp. 345–352.

Degtyarenko, K., de Matos, P., Ennis, M., Hastings, J., Zbinden, M., McNaught, A., Alcntara, R., Darsow, M., Guedj, M. and M.Ashburner (2008). ChEBI: a database and ontology for chemical entities of biological interest, *Nucleic Acids Research* **36**: D344–D350.

D'haeseleer, P., Liang, S. and Somogyi, R. (2000). Genetic network inference: from co-expression clustering to reverse engineering, *Bioinformatics* **16**: 707–726.

D'haeseleer, P., Wen, X., Fuhrman, S. and Somogyi, R. (1999). Linear modeling of mRNA expression levels during CNS development and injury, *Pacific symposium on biocomputing*, pp. 41–52.

Doniger, S., Salomonis, N., Dahlquist, K. D., Vranizan, K., Lawlor, S. C. and Conklin, B. R. (2003). MAPPFinder: using Gene Ontology and GenMAPP to create a global gene-expression profile from microarray data, *Genome Biology* **4**: R7.

Durbin, R., Eddy, S., Krogh, A. and Mitchison, G. (1998). *Biological sequence analysis*, Cambridge university press, Cambridge.

Durstenfeld, R. (1964). Algorithm 235: Random permutation, *Communications of the ACM* **7**: 420.

Felsenstein, J. (1993). PHYLIP (phylogeny inference package), version 3.5c. Department of Genetics, Univ. of Washington, Seattle.

Feng, D. and Dolittle, R. F. (1987). Progressive Sequence Alignment as a Prerequisite to Correct Phylogenetic Trees, *Journal of Molecular Evolution* **25**: 351–360.

Flannick, J., Novak, A., Srinivasan, B. S., McAdams, H. H. and Batzoglou, S. (2006). Graemlin: General and robust alignment of multiple large interaction networks, *Genome Research* **16**: 1169–1181.

Friedman, N., Linial, M., Nachman, I. and Pe'er, D. (2000). Using Bayesian networks to analyze expression data, *RECOMB*, pp. 127–135.

Galperin, M. Y., Walker, D. R. and Koonin, E. V. (1998). Analogous Enzymes: Independent Inventions in Enzyme Evolution, *Genome Research* **8**: 779–790.

Gamalielsson, J. (2000). *Applicability of a Genetic Algorithm when Optimizing a Planar Inverted F Antenna*, Master's thesis, University of Skövde. HS-IDA-EA-00-202.

Gamalielsson, J. (2001). *Models for Protein Structure Prediction by Evolutionary Algorithms*, Master's thesis, University of Skövde. HS-IDA-MD-01-005.

Gamalielsson, J. and Olsson, B. (2002). Using Evolutionary Algorithms to Evaluate Simplified Models for Protein Structure Prediction, *in* H. J. Caulfield, S.-H. Chen, H.-D. Cheng, R. .Duro, V. Honavar, E. E. Kerre, M. Lu, M. Grana-Romay, T. K. Shih, D. Ventura, P. Wang and Y. Yang (eds), *Proc. 6:th Joint Conference on Information Sciences (JCIS 2002), USA*, Association for Intelligent Machinery, North Carolina, USA.

Gamalielsson, J. and Olsson, B. (2004). On the (lack of) robustness of gene expression data clustering, *WSEAS Transactions on Biology and Biomedicine* **1**: 198–204.

Gamalielsson, J. and Olsson, B. (2005a). Evaluating Protein Structure Prediction Models with Evolutionary Algorithms, *in* M. Grana, R. Duro, A. d'Anjou and P. P. Wang (eds), *Information Processing with Evolutionary Algorithms - From Industrial Applications to Academic Speculations*, Springer-Verlag.

Gamalielsson, J. and Olsson, B. (2005b). GOSAP: Gene Ontology Based Semantic Alignment of Biological Pathways, *Technical Report HS-IKI-TR-05-005*, Department of Humanities and Informatics, University of Skövde, Sweden.

Gamalielsson, J. and Olsson, B. (2007). EGOSAP: Evolutionary Gene Ontology based Semantic Alignment of Biological Pathways, *Proc. 3rd Moscow Conference on Computational Molecular Biology (MCCMB07), Moscow, Russia, July 27-30,2007*.

Gamalielsson, J. and Olsson, B. (2008a). Gene Ontology-based Semantic Alignment of Biological Pathways by Evolutionary Search, *Journal of Bioinformatics and Computational Biology (JBCB)* **6**: 825–842.

Gamalielsson, J. and Olsson, B. (2008b). GOSAP: Gene Ontology-based Semantic Alignment of Biological Pathways, *International Journal of Bioinformatics Research and Applications (IJBRA)* **4**: 274–294.

Gamalielsson, J., Nilsson, P. and Olsson, B. (2006). A GO-based Method for Assessing the Biological Plausibility of Regulatory Hypotheses, *in* V. N. Alexandrov, G. D. van Albada, M. A. Sloot and J. Dongarra (eds), *Proceedings of ICCS 2006: 6th International Conference on Computational Science*, Springer-Verlag, pp. LNCS 3992: 879–886.

Gamalielsson, J., Olsson, B. and Nilsson, P. (2005). A Gene Ontology based Method for Assessing the Biological Plausibility of Regulatory Hypotheses, *Technical Report HS-IKI-TR-05-004*, Department of Humanities and Informatics, University of Skövde, Sweden.

Geier, F., Timmer, J. and Fleck, C. (2007). Reconstructing gene-regulatory networks from time series, knock-out data, and prior knowledge, *BMC Systems Biology* **1**: 11.

Hall, D. L. and Llinas, J. (1997). An introduction to multisensor data fusion, *Proceedings of the IEEE* **85**: 6–23.

Hall, D. L. and Llinas, J. (2000). *Handbook of multisensor data fusion*, CRC Press LLC, 2000 N.W. Corporate Blvd., Boca Raton, Florida.

Hartemink, A., Gifford, D., Jaakkola, T. and Young, R. (2002). Combining Location and Expression Data for Principled Discovery of Genetic Regulatory Networks, *Pacific symposium on biocomputing*, pp. 437–449.

Hartemink, A. J. (2005). Reverse engineering gene regulatory networks, *Nature Biotechnology* **23**: 554–555.

Haverty, P. M., Hansen, U. and Weng, Z. (2004). Computational inference of transcriptional regulatory networks from expression profiling and transcription factor binding site identification, *Nucleic Acids Research* **32**: 179–188.

Henikoff, S. and Henikoff, J. (1992). Amino acid substitution matrices from protein blocks, *PNAS* **89**: 10915–10919.

Hochuli, M., Palzelt, M., Oesterhelt, D., Wuthurich, K. and Szypersky, T. (1999). Amino acid biosynthesis in the Halophilic Archaeon Haloarcula Hispanica, *Journal of Bacteriology* **81**: 3226–3237.

Holm, L. and Sander, C. (1996). Mapping the protein universe, *Science* **237**: 595–603.

Husmeier, D. (2003). Sensitivity and specificity of inferring genetic regulatory interactions from microarray experiments with dynamic Bayesian networks, *Bioinformatics* **19**: 2271–2282.

Imoto, S., Savoie, C. J., Aburatani, S., Kim, S., Tashiro, K., Kuhara, S. and Miyano, S. (2003). Use of gene networks for identifying and validating drug targets, *Journal of Bioinformatics and Computational Biology* **1**: 459–474.

Jacob, F. (1977). Evolution and tinkering, *Science* **196**: 1161–1166.

Jiang, J. J. and Conrath, D. W. (1997). Semantic similarity based on corpus statistics and lexical taxonomy, *Proceedings of International Conference on Research in Computation Linguistics (ROCLING X), Taiwan*, pp. 19–33.

Kanehisa, M. and Goto, S. (2000). KEGG: Kyoto Encyclopedia of Genes and Genomes, *Nucleic Acids Research* **28**: 27–30.

Karlsson, S. (2006). *Expression analysis of genes involved in the development of endometrial adenocarcinoma in rat model of human cancer*, PhD thesis, Göteborg University.

Karp, P., Arnaud, M., Collado-Vides, J., Ingraham, J., Paulsen, I. T. and Saier, M. H. J. (2004). The E. coli EcoCyc Database: No Longer Just a Metabolic Pathway Database, *ASM News* **70**: 25–30.

Karp, P. D., Ouzounis, C. A., Moore-Kochlacs, C., Goldovsky, L., Kaipa, P., Ahren, D., Tsoka, S., Darzentas, N., Kunin, V. and Lopez-Bigas, N. (2005). Expansion of the BioCyc collection of pathway/genome databases to 160 genomes, *Nucleic Acids Research* **33**: 6083–6089.

Karp, P. D., Paley, S. and Romero, P. (2002). The pathway tools software, *Bioinformatics* **18**: S1–S8.

Kelley, B. P., Sharan, R., Karp, R. M., Sittler, T., Root, D. E. and Stockwell, B. R. (2003). Conserved pathways within bacteria and yeast as revealed by global protein network alignment, *PNAS* **100**: 11394–11399.

Kelley, B. P., Yuan, B., Lewitter, F., Sharan, R., Stockwell, B. R. and Ideker, T. (2004). PathBLAST: a tool for alignment of protein interaction networks, *Nucleic Acids Research* **32**: W83–W88.

Kim, S. Y., Imoto, S. and Miyano, S. (2003). Inferring gene networks from time series microarray data using dynamic Bayesian networks, *Briefings in Bioinformatics* **4**: 228–235.

Kirschner, M. W. (2005). The Meaning of Systems Biology, *Cell* **121**: 503–504.

Klipcan, L. and Safro, M. (2004). Amino acid biogenesis, evolution of the genetic code and aminoacyl-trna synthases, *Journal of Theoretical Biology* **228**: 389–396.

Knudsen, S. (2002). *A biologist's guide to analysis of DNA microarray data*, Wiley, New York.

Koyutürk, M., Grama, M. and Szpankowski, W. (2004). An efficient algorithm for detecting frequent subgraphs in biological networks, *Bioinformatics* **20**: i200–i207.

Koyutürk, M., Grama, M. and Szpankowski, W. (2005). Pairwise Local Alignment of Protein Interaction Networks Guided by Models of Evolution, *RECOMB*, pp. 48–65.

Lee, T. I., Rinaldi, N. J., Robert, F., Odom, D. T., Bar-Joseph, Z., Gerber, G. K., Hannett, N. M., Harbison, C. T., Thompson, C. M., Simon, I., Zeitlinger, J., Jennings, E. G., Murray, H. L., Gordon, D. B., Ren, B., Wyrick, J. J., Tagne, J.-B., Volkert, T. L., Fraenkel, E., Gifford, D. K. and Young, R. A. (2002). Transcriptional Regulatory Networks in Saccharomyces cerevisiae, *Science* **298**: 799–804.

Liang, S., Fuhrman, S. and Somogyi, R. (1998). REVEAL, a general reverse engineering algorithm for inference of genetic network architectures, *Pacific symposium on biocomputing*, pp. 18–29.

Lin, D. (1998). An information-theoretic definition of similarity, *Proc. of the 15th international conference on machine learning*, Morgan Kaufmann, San Francisco, CA, pp. 296–304.

Liu, B. and Hsu, W. (1996). Post-Analysis of Learned Rules, *AAAI/IAAI, Vol. 1*, pp. 828–834.

Liu, E. T. (2005). Systems Biology, Integrative Biology, Predictive Biology, *Cell* **121**: 505–506.

Lord, P. W., Stevens, R. D., Brass, A. and Goble, C. A. (2003a). Investigating semantic similarity measures across the Gene Ontology: the relationship between sequence and annotation, *Bioinformatics* **19**: 1275–1283.

Lord, P. W., Stevens, R. D., Brass, A. and Goble, C. A. (2003b). Semantic similarity measures as tools for exploring the gene ontology, *Pacific symposium on biocomputing*, pp. 601–612.

Lubovac, Z., Corne, D., Gamalielsson, J. and Olsson, B. (2007). Weighted Cohesiveness for Identification of Functional Modules and their Interconnectivity, *Proc. of the first International Conference on Bioinformatics Research and Development, Berlin, Germany.*

Lubovac, Z., Gamalielsson, J. and Olsson, B. (2006a). Combining functional and topological properties to identify core modules in protein interaction networks, *PROTEINS: Structure, Function and Bioinformatics* **64**: 948–959.

Lubovac, Z., Gamalielsson, J., Olsson, B. and Lindlöf, A. (2005a). Exploring protein networks with a semantic similarity measure, *Proc. 6:th International Symposium on Computational Biology and Genome Informatics (CBGI 2005), USA.*

Lubovac, Z., Olsson, B. and Gamalielsson, J. (2005b). Combining topological properties and domain knowledge reveals functional modules in protein interaction networks, *Proc. 2:nd International Conference on Algorithms and Computational Methods for Biochemical and Evolutionary Networks (CompBioNets 2005), Lyon, France*.

Lubovac, Z., Olsson, B. and Gamalielsson, J. (2006b). Weighted Clustering Coefficient for Identifying Modular Formations in Protein-Protein Interaction Networks, *Proc. 3:rd International Conference on Bioinformatics and Computational and Systems Biology, Prague, Czech republic*.

Ma, H. and Horiuchi, K. Y. (2006). Chemical microarray: a new tool for drug screening and discovery, *Drug Discovery Today* **11**: 661–668.

MacBeath, G. and Schreiber, S. L. (2000). Printing Proteins as Microarrays for High-Throughput Function Determination, *Science* **289**: 1760–1763.

Margulis, L. and Schwartz, K. V. (1997). *Five Kingdoms: An Illustrated Guide to the Phyla of Life on Earth*, W.H. Freeman & Company, San Francisco.

Maslov, S. and Sneppen, K. (2002). Specificity and Stability in Topology of Protein Networks, *Science* **296**: 910–913.

McLachlan, A. D. (1982). Rapid comparison of protein structures, *Acta Crystallographica* **A38**: 871–873, as implemented in the program ProFit (Martin, A.C.R., http://www.bioinf.org.uk/software/profit/).

Michalewicz, Z. and Fogel, D. B. (2004). *How to Solve It: Modern Heuristics, Second edition*, Springer, Berlin.

Miller, G. A. (1990). Wordnet: an on-line lexical database, *International Journal of Lexicography* **3**: 235–312.

Mootha, V. K., Lindgren, C. M., Eriksson, K.-F., Subramanian, A., Sihag, S., Lehar, J., Puigserver, P., Carlsson, E., Ridderstrale, M., Laurila, E., Houstis, N., Daly, M. J., Patterson, N., Mesirov, J. P., Golub, T. R., Tamayo, P., Spiegelman, B., Lander, E. S., Hirschhorn, J. N., Altschuler, D. and Groop, L. C. (2003). Pgc-1 alpha-responsive genes involved in oxidative phosphorylation are coordinately downregulated in human diabetes, *Nature genetics* **34**: 267–273.

Morin, P. J. (1999). *Community Ecology*, Blackwell Science, Malden, Massachusetts.

Narayanan, A., Cheung, A., Gamalielsson, J., Keedwell, E. and Vercellone, C. (2005). ANNs and GAs for Reducing the Dimensionality of Gene Expression Data, *in* U. Seiffert, L. C. Jain and P. Schweizer (eds), *Bioinformatics using Computational Intelligence Paradigms, Studies in Fuzziness and Soft Computing*, Springer-Verlag, pp. 191–216.

Narayanan, A., Keedwell, E., Gamalielsson, J. and Tatineni, J. (2004). Single Layer Artificial Neural Networks for Gene Expression Analysis, *Neurocomputing* **61**: 214–240.

Needleman, S. and Wunsch, C. (1970). A general method applicable to the search for similarities in the amino acid sequence of two proteins, *Journal of Molecular Biology* **48**: 443–453.

Nilsson, E. C., Long, Y. C., Martinsson, S., Glund, S., Garcia-Roves, P., Svensson, T. L., Andersson, L., Zierath, J. R. and Mahlapuu, M. (2006). Opposite transcriptional regulation in skeletal muscle of AMPK γ3 R225Q transgenic versus knock-out mice, *Journal of Biological Chemistry* **281**: 7244–7252.

Nomenclature Committee of the International Union of Biochemistry and Molecular Biology (1992). *Enzyme Nomenclature*, Academic Press, San Diego.

Ogata, H., Fujibuchi, W., Goto, S. and Kanehisa, M. (2000). A heuristic graph comparison algorithm and its application to detect functionally related enzyme clusters, *Nucleic Acids Research* **28**: 4021–4028.

Olsson, B., Gawronska, B. and Erlendsson, B. (2006). Deriving pathway maps from text analysis using a grammar-based approach, *Journal of Bioinformatics and Computational Biology (JBCB)* **4**: 483–502.

Oltvai, Z. N. and Barabasi, A.-L. (2002). Life's Complexity Pyramid, *Science* **298**: 763–764.

Ortiz, A. R., Strauss, C. E. and Olmea, O. (2002). MAMMOTH (Matching molecular models obtained from theory): An automated method for model comparison, *Protein Science* **11**: 2606–2621.

Padmanabhan, B. and Tuzhilin, A. (1999). Unexpectedness as a measure of interestingness in knowledge discovery, *Decision Support Systems* **27**: 303–318.

Pennisi, E. (2007). Working the (Gene Count) Numbers: Finally, a Firm Answer, *Science* **316**: 1113.

Pinter, R. Y., Rokhlenko, O., Yeger-Lotem, E. and Ziv-Ukelson, M. (2005). Alignment of metabolic pathways, *Bioinformatics* **21**: 3401–3408.

Pruitt, K. D., Tatusova, T. and Maglott, D. R. (2007). NCBI Reference Sequence (RefSeq): a curated non-redundant sequence database of genomes, transcripts and proteins, *Nucleic Acids Research* **35**: D61–D65.

Resnik, P. (1999). Semantic Similarity in a Taxonomy: An Information-Based Measure and its Application to Problems of Ambiguity in Natural Language, *Journal of Artificial Intelligence Research (JAIR)* **11**: 95–130.

Robbins, H. (1952). Some Aspects of the Sequential Design of Experiments, *Bulletin of the American Mathematical Society* **58**: 527–535.

Rosen, K. H. (1995). *Discrete mathematics and its applications, 3rd edition*, McGraw Hill, New York.

Rowell, C. and Gamalielsson, J. (1997). CGAMAS: a post-processing tool for microwave data files, *IEE Colloquium on Effective Microwave CAD (Ref. No: 1997/377)* pp. 4/1–4/4.

Russell, S. and Norvig, P. (1995). *Artificial Intelligence - a modern approach*, Prentice-Hall, New Jersey.

Schena, M., Shalon, D., Davis, R. W. and Brown, P. O. (1995). Quantitative monitoring of gene expression patterns with a complementary DNA microarray, *Science* **270**: 467–470.

Sharan, R., Suthram, S., Kelley, R. M., Kuhn, T., McCuine, S., Uetz, P., Sittler, T., Karp, R. M. and Ideker, T. (2005). Conserved patterns of protein interaction in multiple species, *PNAS* **102**: 1974–1979.

Shen-Orr, S. S., Milo, R., Mangan, S. and Alon, U. (2002). Network Motifs in the transcriptional regulation network of Escherichia coli, *Nature genetics* **31**: 64–68.

Shlomi, T., Segal, D., Ruppin, E. and Sharan, R. (2006). QPath: a method for querying pathways in a protein-protein interaction network, *BMC Bioinformatics* **7**: 199–207.

Smith, B., Ashburner, M., Rosse, C., Bard, J., Bug, W., Ceusters, W., Goldberg, L. J., Eilbeck, K., Ireland, A., Mungall, C. J., Leontis, N., Rocca-Serra, P., Ruttenberg, A., Sansone, S.-A., Scheuermann, R. H., Shah, N., Whetzel, P. L. and Lewis, S. (2007). The OBO Foundry: coordinated evolution of ontologies to support biomedical data integration, *Nature Biotechnology* **25**: 1251–1255.

Smith, T. F. and Waterman, M. S. (1981). Identification of Common Molecular Subsequences, *Journal of Molecular Biology* **147**: 195–197.

Speer, N., Spieth, C. and Zell, A. (2004). A Memetic Co-Clustering Algorithm for Gene Expression Profiles and Biological Annotation, *Proc. of the 2004 Congress on Evolutionary Computation, Portland, Oregon, USA, June 19-23*, IEEE Press, pp. 1631–1638.

Steinberg, A. N., Bowman, C. L. and White, F. E. (1999). Revisions to the JDL Data Fusion Model, *Proc. of the SPIE Conference on Sensor Fusion: Architectures, Algorithms, and Applications, Orlando, Florida, April 1999*, pp. Vol. 3719, 277–786X/99.

Storey, J. D. and Tibshirani, R. (2003). Statistical significance for genomewide studies, *PNAS* **100**: 9440–9445.

Subramanian, A., Tamayo, P., Mootha, V. K., Mukherjee, S., Ebert, B. L., Gillette, M. A., Paulovich, A., Pomeroy, S. L., Golub, T. R., Lander, E. S. and Mesirov, J. P. (2005). Gene set enrichment analysis: A knowledge-based approach for interpreting genome-wide expression profiles, *PNAS* **102**: 15545–15550.

Swets, J. A., Dawes, R. M. and Monahan, J. (2000). Psychological science can improve diagnostic decisions, *Pshychological science in the public interest* **1**: 1–26.

Synnergren, J., Gamalielsson, J. and Olsson, B. (2007). Mapping of the JDL Data Fusion Model to Bioinformatics, *Proc. 2007 IEEE International Conference on Systems, Man and Cybernetics (SMC 2007), Montreal, Canada, October 7-10, 2007.*

Synnergren, J., Olsson, B. and Gamalielsson, J. (2008). A data integration method for exploring gene regulatory mechanisms, *Proc. ACM 17th Conference on Information and Knowledge Management (CIKM 2008), Napa Valley, California, October 26-30, 2008.*

Tatusova, T. A. and Madden, T. L. (1999). BLAST 2 SEQUENCES, a new tool for comparing protein and nucleotide sequences, *FEMS Microbiology Letters* **147**: 247–250.

Tian, Y., McEachin, R. C., Santos, C., States, D. J. and Patel, J. M. (2007). SAGA: a subgraph matching tool for biological graphs, *Bioinformatics* **23**: 232–239.

Tohsato, Y., Matsuda, H. and Hashimoto, A. (2000). A multiple alignment algorithm for metabolic pathway analysis using enzyme hierarchy, *Proc. 8th International Conference on Intelligent Systems for Molecular Biology (ISMB 2000)*, pp. 376–383.

Tuzhilin, A. and Adomavicius, G. (2002). Handling Very Large Numbers of Association Rules in the Analysis of Microarray Data, *Knowledge Discovery and Data Mining*, pp. 396–404.

Wasserman, S. and Faust, K. (1994). *Social Network Analysis*, Cambridge University Press, Cambridge.

Weaver, D. C., Workman, C. T. and Stormo, G. D. (1999). Modeling regulatory networks with weight matrices, *Pacific symposium on biocomputing*, pp. 112–123.

Werhli, A. D., Grzegorczyk, M. and Husmeier, D. (2006). Comparative evaluation of reverse engineering gene regulatory networks with relevance networks, graphical Gaussian models and Bayesian networks, *Bioinformatics* **22**: 2523–2531.

Wernicke, S. and Rasche, F. (2007). Simple and fast alignment of metabolic pathways by exploiting local diversity, *Bioinformatics* **23**: 1978–1985.

Wolpert, D. H. and Macready, W. G. (1997). No Free Lunch Theorems for Optimization, *IEEE Transactions on Evolutionary Computation* **1**: 67–82.

Yang, Q. and Sze, S.-H. (2007). Path Matching and Graph Matching in Biological Networks, *Journal of Computational Biology* **14**: 56–67.

Zhou, X., Wang, X., Pal, R., Ivanov, I., Bittner, M. and Dougherty, E. R. (2004). A Bayesian connectivity-based approach to constructing probabilistic gene regulatory networks, *Bioinformatics* **20**: 2918–2927.

Zou, M. and Conzen, S. D. (2005). A new dynamic Bayesian network (DBN) approach for identifying gene regulatory networks from time course microarray data, *Bioinformatics* **21**: 71–79.